# required R packages:

# requires (lme4)

# requires (afex)

# requires (tidyverse)

# requires (psych)

# requires (optimix)

# requires (ggpubr)

# requires (broom.mixed)

setwd("C:/Rfiles") # sets folder

rm(list = ls ()) # clears R's memory

binarydata <- read.csv("peerj-74911-data.csv", na.strings = c("","NA"), header=TRUE) # opens file

set\_sum\_contrasts() # criterion for running the contrasts in afex

View(binarydata) # view dataset

str(binarydata) # summary of the types of variables and their values

# changes variables to factor vars

binarydata$species <- as.factor(binarydata$species)

# making dependent vars numerical factors

binarydata$collnear <- as.numeric(as.character(binarydata$collnear))

binarydata$collafter <- as.numeric(as.character(binarydata$collafter))

# absolute values for horizontal angle

binarydata <- mutate(binarydata, horizangleabs = abs(horizangle - 90))

#correlations between independent continuous factors in the dataset

ind.cont1 <- binarydata[c("horizangleabs", "time\_dec", "velocity")]

corr.test(ind.cont1, use = "pairwise", method = "pearson", adjust = "none")

# no high correlation coefficients so we include them all

str(binarydata)

# assessing normality of independent continuous factors

qqnorm(binarydata$velocity)

qqline(binarydata$velocity)

qqnorm(binarydata$horizangleabs)

qqline(binarydata$horizangleabs)

qqnorm(binarydata$time\_dec)

qqline(binarydata$time\_dec)

# providing descriptive statistics on the independent factors

descr(binarydata$velocity, style="rmarkdown")

descr(binarydata$horizangleabs, style="rmarkdown")

descr(binarydata$time\_dec, style="rmarkdown")

#Centering the continuous independent variable dist.from.media

MyNorm <- function(x){ (x-mean(x, na.rm = TRUE))/sd(x, na.rm = TRUE)}

#Add na.rm = TRUE to deal with datasets with NA's

binarydata$velocityc <- MyNorm(binarydata$velocity)

binarydata$horizangleabsc <- MyNorm(binarydata$horizangleabs)

binarydata$time\_decc <- MyNorm(binarydata$time\_dec)

###### Dependent variable: probability of collision with window when birds flew close to the window

# model with centered variables to obtain effect sizes (odds ratios)

collnearm1 <- mixed(collnear ~ velocityc + horizangleabsc + time\_decc + (1|species), data = binarydata, method = "LRT", family = binomial, expand\_re = TRUE)

anova(collnearm1)

# model has convergence issues

# we assess the potential convergence issues in the following lines

#### https://rstudio-pubs-static.s3.amazonaws.com/33653\_57fc7b8e5d484c909b615d8633c01d51.html

# we check for singularity issues

tt <- getME(collnearm1$full\_model,"theta")

ll <- getME(collnearm1$full\_model,"lower")

min(tt[ll==0])

# singularity does not seem to be a problem because theta is much hight than zero

# we increase the number of iterations of the model

ss <- getME(collnearm1$full\_model,c("theta","fixef"))

m2 <- update(collnearm1$full\_model,start=ss,control=glmerControl(optCtrl=list(maxfun=2e4)))

# now there are no convergence issues

##### we try different optimizers following this approach:

## https://rstudio-pubs-static.s3.amazonaws.com/33653\_57fc7b8e5d484c909b615d8633c01d51.html

aa <- allFit(collnearm1$full\_model)

is.OK <- sapply(aa,is,"merMod")

aa.OK <- aa[is.OK]

lapply(aa.OK,function(x) x@optinfo$conv$lme4$messages)

# several optimizers led to the convergence of the model (NULL)

# we try re-running the model with the following optimizer:nlminbwrap

collnearm2 <- mixed(collnear ~ velocityc + horizangleabsc + time\_decc + (1|species), data = binarydata, method = "LRT", family = binomial, control = glmerControl(optimizer = "nlminbwrap"), expand\_re = TRUE)

anova(collnearm2)

# no issues of convergence with the nlminbwrap optimizer

##### we try different optimizers following this approach:

## https://joshua-nugent.github.io/allFit/

aa <- allFit(collnearm1$full\_model)

is.OK <- sapply(aa,is,"merMod")

aa.OK <- aa[is.OK]

lapply(aa.OK,function(x) x@optinfo$conv$lme4$messages)

diff\_optims <- allFit(collnearm1$full\_model, maxfun = 1e5)

diff\_optims\_OK <- diff\_optims[sapply(diff\_optims, is, "merMod")]

lapply(diff\_optims\_OK, function(x) x@optinfo$conv$lme4$messages)

convergence\_results <- lapply(diff\_optims\_OK, function(x) x@optinfo$conv$lme4$messages)

working\_indices <- sapply(convergence\_results, is.null)

if(sum(working\_indices) == 0){

 print("No algorithms from allFit converged. You may still be able to use the results, but proceed with extreme caution.")

 first\_fit <- NULL

} else {

 first\_fit <- diff\_optims[working\_indices][[1]]

}

first\_fit

# the parameter estimates are the same as obtained following the first approach:

summary (collnearm2$full\_model)

# attributes of model object

attributes (collnearm2$full\_model)

#### parameters in odds ratio scales for effect size reporting

tidy(collnearm2$full\_model,conf.int=TRUE,exponentiate=TRUE,effects="fixed")

# Figures

# plotting velocity effects

ggplot(binarydata, aes(x = velocity, y = collnear)) + geom\_point() +

 stat\_smooth(method="glm", method.args=list(family="binomial"), se = TRUE) +

 labs(y = "Probability of collision", x = "Velocity (m/s)") + theme\_pubr(16) +

 scale\_x\_continuous(breaks=seq(2,12,1))

# plotting angle effects

ggplot(binarydata, aes(x = horizangleabs, y = collnear)) + geom\_point() +

 stat\_smooth(method="glm", method.args=list(family="binomial"), se = TRUE) +

 labs(y = "Probability of collision", x = "Horizontal angle of approach (°)") + theme\_pubr(16) +

 scale\_x\_continuous(breaks=seq(0,50,5))

# plotting time

ggplot(binarydata, aes(x = time\_dec, y = collnear)) + geom\_point() +

 stat\_smooth(method="glm", method.args=list(family="binomial"), se = TRUE) +

 labs(y = "Probability of collision", x = "Time of the day") + theme\_pubr(16) +

 scale\_x\_continuous(breaks=seq(2,22,2))

###### Dependent variable: probability of visible injury (stunned, died) after birds collided with window

collafterm1 <- mixed(collafter ~ velocityc + horizangleabsc + time\_decc + (1|species), data = binarydata, method = "LRT", family = binomial, expand\_re = TRUE)

anova(collafterm1)

# we confirm the singularity issue

tt <- getME(collafterm1$full\_model,"theta")

ll <- getME(collafterm1$full\_model,"lower")

min(tt[ll==0])

# singularity confirmed being zero

# we increase the number of iterations of the model

ss <- getME(collafterm1$full\_model,c("theta","fixef"))

m2 <- update(collafterm1$full\_model,start=ss,control=glmerControl(optCtrl=list(maxfun=2e4)))

# singularity persists

##### we try different optimizers following this approach:

## https://rstudio-pubs-static.s3.amazonaws.com/33653\_57fc7b8e5d484c909b615d8633c01d51.html

aa <- allFit(collafterm1$full\_model)

is.OK <- sapply(aa,is,"merMod")

aa.OK <- aa[is.OK]

lapply(aa.OK,function(x) x@optinfo$conv$lme4$messages)

# singularity persists

# singularity should not be caused by collinearity between the independent factors

# as they are not correlated (see above)

# we subset the data to only species with more than one entry,

# which might improve the estimation of the random variance

binarydata1 <- binarydata [which(binarydata$species == "Blue jay" | binarydata$species == "Cardinal" | binarydata$species == "Downy woodpecker" | binarydata$species == "Grackle" | binarydata$species == "Grosbeak"),]

#### we run the mode with more than 1 entry per species

collaftermsubset <- mixed(collafter ~ velocityc + horizangleabsc + time\_decc + (1|species), data = binarydata1, method = "LRT", family = binomial, expand\_re = TRUE)

anova(collaftermsubset)

# singularity persists

### we run the original model (with all species) with individual factors

collafterm\_v <- mixed(collafter ~ velocityc + (1|species), data = binarydata, method = "LRT", family = binomial, expand\_re = TRUE)

anova(collafterm\_v)

collafterm\_a <- mixed(collafter ~ horizangleabsc + (1|species), data = binarydata, method = "LRT", family = binomial, expand\_re = TRUE)

anova(collafterm\_a)

collafterm\_t <- mixed(collafter ~ time\_decc + (1|species), data = binarydata, method = "LRT", family = binomial, expand\_re = TRUE)

anova(collafterm\_t)

# the singularity went away from one of the single-independent factor models

# most importantly, we confirmed that both velociy and horizangleabs yielded the same significant results

# in the single-independent factor versions of the models.

summary (collafterm1)

#### parameters in odds ratio scales for effect size reporting

tidy(collafterm1$full\_model,conf.int=TRUE,exponentiate=TRUE,effects="fixed")

# Figures

# plotting velocity effects

ggplot(binarydata, aes(x = velocity, y = collafter)) + geom\_point() +

 stat\_smooth(method="glm", method.args=list(family="binomial"), se = TRUE) +

 labs(y = "Probability of visible injury", x = "Velocity (m/s)") + theme\_pubr(16) +

 scale\_x\_continuous(breaks=seq(2,12,1))

# plotting angle effects

ggplot(binarydata, aes(x = horizangleabs, y = collafter)) + geom\_point() +

 stat\_smooth(method="glm", method.args=list(family="binomial"), se = TRUE) +

 labs(y = "Probability of visible injury", x = "Horizontal angle of approach (°)") + theme\_pubr(16) +

 scale\_x\_continuous(breaks=seq(0,50,5))

# plotting time

ggplot(binarydata, aes(x = time\_dec, y = collafter)) + geom\_point() +

 stat\_smooth(method="glm", method.args=list(family="binomial"), se = TRUE) +

 labs(y = "Probability of visible injury", x = "Time of the day") + theme\_pubr(16) +

 scale\_x\_continuous(breaks=seq(2,22,2))