#################################################################

#################################################################

############# CLIMATE CHANGE AND VIOLENCE SCRIPT ###########

#################################################################

#################################################################

### Thomas J. Snyder

### 10/19/2022

#### Loading required library packages ####

{library(lme4)

 library(lmerTest)

 library(lawstat)

 library(dunn.test)

 library(ggeffects)

 library(ggplot2)

 library(tidyr)

 library(FSA)

}

#### Importing and Inspecting Climate Data #####

par(mfrow=c(1,1)) #Set graphing parameters

quelccaya <- read.csv('') #Imports Moseley-Thompson 1992 climate data from Quelccaya glacier; snow acc. and ice d18O

head(quelccaya) #visually inspects imported data

plot(quelccaya$accumulation~quelccaya$year,type='l') #creates lineplot of Quelccaya glacier yearly ice accumulation

### Plotting Quelccaya data, with archaeological periods clear ###

plot(quelccaya$accumulation~quelccaya$year, cex=0, xlab = 'Year CE',bty = "l",

 ylab = 'Annual Ice Accumulation (m)',

 ylim= c(0.75,2)) #creates empty plot with

rect(xleft=400,xright = 600, ybottom = 0, ytop = 3, density=8, col = "orange")

rect(xleft=600,xright = 1000, ybottom = 0, ytop = 3, density=8, col = "blue")

rect(xleft=1000,xright = 1400, ybottom = 0, ytop = 3, density=8, col = "orange")

rect(xleft=1400,xright = 1700, ybottom = 0, ytop = 3, density=8, col = "blue")

abline(h = mean(quelccaya$accumulation), col = 'black', lwd=3, lty=3) # draws horizontal black line at overall mean precipitation value

lines(quelccaya$year, predict(loess(accumulation~as.numeric(year), span=0.05,

 data=quelccaya)), lwd=2, col='red', )

text(525, 0.8, 'EIP')

text(800, 0.8, 'MH', col='white', cex = 1.3)

text(800, 0.8, 'MH')

text(1200, 0.8, 'LIP', col='white', cex=1.3)

text(1200, 0.8, 'LIP')

text(1500, 0.8, 'LH', col='white', cex=1.3)

text(1500, 0.8, 'LH')

# Calculating Basic stats precipitation by archaeological period #

tapply(quelccaya$accumulation, quelccaya$period, mean) #observes average yearly snow accumulation Quellcaya by archaeological period

tapply(quelccaya$accumulation, quelccaya$period, sd)#observes standard deviation of yearly snow accumulation Quellcaya by archaeological period

# Tests for normality, see if ANOVA is appropriate (it isn't, need to use non-parametric) #

tapply(quelccaya$accumulation, quelccaya$period, shapiro.test)

tapply(quelccaya$accumulation, quelccaya$period, shapiro.test)

# Kruskal Wallis for snow accumulation by period #

kwptest <- kruskal.test(quelccaya$accumulation~quelccaya$period) #performs Kruskal-Wallis test for difference between two non-normally distributed samples

kwptest #returns results

dtp <- dunn.test(quelccaya$accumulation~quelccaya$period) #Dunn's Test of difference for statistically significant Kruskal-Wallis values

dtp #returns results

############# Observed differences: snow accumulation between LH/LIP;MH/LIP'############

# Boxplot by archaeological period #

periods <- as.factor(quelccaya$period) #sets the archaeological period as factors

periods <- factor(periods, levels = c('eip','mh','lip', 'lh'), ordered = TRUE) #puts archaeological periods in archaeological order

boxplot(quelccaya$accumulation~periods, col = c('orange', 'green', 'orange', 'green'), xlab='Archaeological Periods', ylab='Decadal Average Ice Accumulation') #plots average precipitation per archaeological period

abline(h=mean(quelccaya$accumulation), col='red', lty = 'dashed', lwd = 3) #draws midline on previous graph

#### Violence Data ######

### Imports violence data (cranial trauma as proxy, walker 2001), collected from 29 articles to date; not all data fits quelccaya

violence <- read.csv('') #Imports violence data

head(violence) #visually inspects for correct importation

violence$periods <- as.factor(violence$period) #sets archaeological period as a factor

violence$periods <- factor(violence$periods, levels = c('eip','mh','lip', 'lh'), ordered = TRUE)#puts

# Basic stats for violence #

tapply(violence$percent, violence$period, FUN = mean) #mean values for violence by archaeological period

tapply(violence$percent, violence$period, FUN = quantile) #mean values for violence by archaeological period

eip4quant <- subset(violence, violence$period=='eip')

quantile(eip4quant$percent,probs=c(.025,.975))

mh4quant <- subset(violence, violence$period=='mh')

quantile(mh4quant$percent,probs=c(.025,.975))

lip4quant <- subset(violence, violence$period=='lip')

quantile(lip4quant$percent,probs=c(.025,.975))

lh4quant <- subset(violence, violence$period=='lh')

quantile(lh4quant$percent,probs=c(.025,.975))

# Tests for normality (fails) #

tapply(violence$percent, violence$period, shapiro.test) #tests for normality

# Kruskal-Wallis Violence

kwv <- kruskal.test(violence$percent~violence$period) #tests for differences in rates of overall violence by archaeological period with the Kruskalis Wallis test

kwv #does not detect significant differences

# Kruskal-Wallis Environment #

environment <- as.factor(violence$environment) #sets environment as a factor

environment <- factor(environment, levels = c('coast', 'midland', 'highland'), ordered = TRUE) ## 0-500 = Coast, 500-3000 = mid; x > 3000 = high | Following Wilson et al. 2022

kwe <- kruskal.test(violence$percent~environment) #tests for differences in rates of overall violence between elevations (Not significant)

kwe #Kruskal Wallis results

# Boxplot of violence by Archaeological Period #

boxplot(violence$percent~violence$periods, col = c('orange', 'green', 'orange', 'green'), xlab = "Archaeological Period", ylab = "Percent of Cranial Population with Traumatic Lesions") #boxplot of violence by archaeological material

abline(h = mean(violence$percent), col = 'red') # midline

# Boxplot by environment #

boxplot(violence$percent~environment, main = "Violence by Environmental Region", xlab ='Environment', ylab = 'Percent of Cranial Population with at least one Traumatic Lesion', col = c('cyan', 'green', 'orange')) #Violence v. Environmental region

#### Binomial glm analyses, broad #####

violence <- read.csv('') ## Subset of trauma data for which there's data for Quelccaya from

#calculating the empirical odds of trauma

violence$emp.odds <- (violence$trauma+0.5)/(violence$notrauma+0.5) #corrects for 0's, calculates empirical odds of violence per observation

violence$emp.odds #returns empirical odds of violence

#this is a scatter plot of the log of the empirical odds versus the mean width

#we make an empty plot region first

plot(x = violence$mean\_precipitation, y = log(violence$emp.odds), type ="n", bty = "n", xlab="",ylab="",xaxt="n",yaxt="n") #creates an empty graph for filling in later - without labels, axes, or the big box

symbols(x = violence$mean\_precipitation, y = log(violence$emp.odds), circles = sqrt(violence$observed), inches=0.2, add=TRUE) #adds in sumbols - x axis is mean precipitation, y is log odds, circles are proportional to sample size

lines(smooth.spline(x = violence$mean\_precipitation, y = log(violence$emp.odds), spar= 1), col = 'blue',lwd=3)#adds in a line of best fit

axis(side=1) ##adds an x axis, meters of ice accumulate

axis(side=2,at=c(log(1/10),log(1/5),log(0.5),log(1),log(2)),labels=c("1:10","1:5","1:2","1:1","2:1")) #adds a y axis, log odds of violence

title(xlab="meters of ice accumulate", ylab = "Empirical odds of trauma (log scale)") #adds axis labesls

#### GLM model additive ####

fitadditive <- glmer(cbind(trauma,notrauma)~mean\_precipitation+environment+(1|site),data=violence,nAGQ=15,family=binomial) #additive model of precipitation and environment

summary(fitadditive) #values from additive

#### GLM model interaction \*\*\*\* this one \*\*\*\*\* not fishing ####

fitinteraction <- glmer(cbind(trauma,notrauma)~mean\_precipitation\*environment+(1|site),data=violence,nAGQ=15,family=binomial) #interactive model, of environment and precipitation

summary(fitinteraction) #summary of additive model

res.pearson <- residuals(fitinteraction,type="pearson") #residuals of additive model

plot(x = log(fitted(fitinteraction)),y=res.pearson,bty="n",xlab="Predicted Count (Log Scale)",ylab= "Pearson Residual") #plot of fitted interaction

sum(res.pearson^2) #measure of fit - very good

plot(fitinteraction) #plots pearsons residuals

#### creating model plot ####

pred.violence <- predict(fitinteraction,type="response") #predicting values based on model

x <- violence %>%

 ggplot(aes(mean\_precipitation,((trauma+1)/(observed+1))))+

 geom\_point(aes(color=environment,shape=environment),size=sqrt(violence$observed\*0.5))+

 geom\_smooth(method = 'glm',method.args=list(family="quasibinomial"),se=TRUE,

 aes(y=pred.violence,group=environment,color=environment,

 fill=environment))+

 theme(legend.position=c(1,1))+

 theme\_minimal()+

 labs(x='Annual Ice Accumulation (meters)', y='Frequency of Violence',

 title = 'Annual Ice Accumulation vs. Frequency of Violence')+

 theme(plot.title = element\_text(hjust = 0.5))+

 scale\_fill\_manual(values = c("grey80","grey80","grey80"),

 breaks = c("coast","midland","highland"),

 name=("Elevational Zone"),

 labels=c("Coastal","Mid-Elevation","Highland"))+

 scale\_shape\_manual(values = c(0,1,2),

 breaks = c("coast","midland","highland"),

 name=("Elevational Zone"),

 labels=c("Coastal","Mid-Elevation","Highland"))+

 scale\_color\_manual(values = c("slateblue","green2","red"),

 breaks = c("coast","midland","highland"),

 name=("Elevational Zone"),

 labels=c("Coastal","Mid-Elevation","Highland"))## GG plot of Corrected Empirical odds of violence vs. annual ice accumulation|using frequency instead of empirical odds due to error message related to predictions being length = 44 and trauma to no trauma matrix being length = 88 | additionally, error when using odds

x#plots ^

#### calculating pearsons residual by hand to check model goodness of fit ####

pred.violence #returns predicted violence values

trauma\_frequency <- (violence$trauma+0.5)/(violence$observed+0.5) #defines frequency of trauma

predicted\_trauma\_count<- pred.violence\*violence$observed #defines predicted trauma count

sti <- sd(pred.violence) #calculates the standard deviation of predicted violence values

glmerpearson <- function(yi,yihat,sdihat){

 #calculates pearson's residual

 #inputs should be yi (in this case frequency of violence in the normal population), yihat (predicted trauma frequencies), sdihat (sd of predicted trauma frequencipes)

 pearsonresidual <- ((yi-yihat)/(sdihat))

 return(pearsonresidual)

} #writes function for determining pearson residuals

y.hat <- violence$observed\*pred.violence #defines y.hat as violence$observed\*predicted violence frequencies

y.hat #returns y.hat

sd.hat <- sqrt(violence$observed\*pred.violence\*(1-pred.violence)) #defines sd.hat as the standard deviation

sd.hat #returns sd.hat

mypearson.res <- (violence$trauma-y.hat)/sd.hat #plugs known values into formula

plot(mypearson.res~res.pearson) #plugs hand-created values into previously existing function to check they're 1:1

#### Effect size estimation- the change in log odds of trauma per unit change in meters of ice accumulation for each environment ####

lowland.bin <- c(0,1,0,0,0,0) #sets matrix for model coefficients relevant for lowlands

highland.bin <- c(0,1,0,0,1,0) #sets matrix for model coefficients relevant for highlands

midland.bin <- c(0,1,0,0,0,1) ##sets matrix for model coefficients relevant for midlands

## these are gonna be the effect sizes in the log odds scaling for a 1 unit change in meters of ice accumulate

lowland.eff <- t(lowland.bin)%\*%coef(summary(fitinteraction),fixed.only=TRUE)[,"Estimate"] #transposes matrix and multiplies by model coefficients

lowland.se <- sqrt(t(lowland.bin)%\*%vcov(fitinteraction)%\*%lowland.bin)

highland.eff <- t(highland.bin)%\*%coef(summary(fitinteraction),fixed.only=TRUE)[,"Estimate"]#transposes matrix and multiplies by model coefficients

highland.se <- sqrt(t(highland.bin)%\*%vcov(fitinteraction)%\*%highland.bin)

midland.eff <- t(midland.bin)%\*%coef(summary(fitinteraction),fixed.only=TRUE)[,"Estimate"]#transposes matrix and multiplies by model coefficients

midland.se <- sqrt(t(midland.bin)%\*%vcov(fitinteraction)%\*%midland.bin)

highland.eff \* 0.1 #returns highland effects per 0.1 meter ice change

exp(0.88) #exponentiates highland.eff

highland.se\*0.1 #returns highland s.e per 0.1 meter ice change

exp(0.88-0.29) #calculates lower 95% bound

exp(0.88+0.29) #calculates upper 95% bound

################### S.D. ####################################################################################

fitinteraction.sd. <- glmer(cbind(trauma,notrauma)~sd\_precipitation\*environment+(1|site),data=violence,nAGQ=15,family=binomial) #interactive model of trauma likelihood as a function of standard deviation of environment and elevational zone

fitinteraction.sd.#returns model

summary(fitinteraction.sd.) #returns summary of model

plot(x = violence$sd\_precipitation, y = log(violence$emp.odds), type ="n", bty = "n", xlab="",ylab="",xaxt="n",yaxt="n") #creates an empty graph for filling in later - without labels, axes, or the big box

symbols(x = violence$sd\_precipitation, y = log(violence$emp.odds), circles = sqrt(violence$observed), inches=0.2, add=TRUE) #adds in sumbols - x axis is mean precipitation, y is log odds, circles are proportional to sample size

lines(smooth.spline(x = violence$sd\_precipitation, y = log(violence$emp.odds), spar= 1), col = 'blue',lwd=3)#adds in a line of best fit

axis(side=1) ##adds an x axis, meters of ice accumulate

axis(side=2,at=c(log(1/10),log(1/5),log(0.5),log(1),log(2)),labels=c("1:10","1:5","1:2","1:1","2:1")) #adds a y axis, log odds of violence

title(xlab="meters of ice accumulate", ylab = "Empirical odds of trauma (log scale)") #adds axis labesls

pred.violence.sd <- predict(fitinteraction.sd.,type='response')#writes predicted values according to the s.d. model

res.pearson.sd <- residuals(fitinteraction.sd.,type="pearson") #returns pearson's resilduals for s.d. model

plot(res.pearson.sd) #plots pearson residuals

sum(res.pearson.sd)^2 #returns sum of squared pearson's residuals as a measure of goodness of fit

y <- violence %>%

 ggplot(aes(sd\_precipitation,((trauma+1)/(observed+1))))+

 geom\_point(aes(color=environment,shape=environment),size=sqrt(violence$observed\*0.5))+

 geom\_smooth(method = 'glm',method.args=list(family="quasibinomial"),se=TRUE,

 aes(y=pred.violence.sd,group=environment,color=environment,

 fill=environment))+

 theme(legend.position=c(1,1))+

 theme\_minimal()+

 labs(x='Annual Ice Accumulation (meters)', y='Frequency of Violence',

 title = 'Annual Ice Accumulation vs. Frequency of Violence')+

 theme(plot.title = element\_text(hjust = 0.5))+

 scale\_fill\_manual(values = c("grey80","grey80","grey80"),

 breaks = c("coast","midland","highland"),

 name=("Elevational Zone"),

 labels=c("Coastal","Mid-Elevation","Highland"))+

 scale\_shape\_manual(values = c(0,1,2),

 breaks = c("coast","midland","highland"),

 name=("Elevational Zone"),

 labels=c("Coastal","Mid-Elevation","Highland"))+

 scale\_color\_manual(values = c("slateblue","green2","red"),

 breaks = c("coast","midland","highland"),

 name=("Elevational Zone"),

 labels=c("Coastal","Mid-Elevation","Highland"))## GG plot of Corrected Empirical odds of violence vs. annual ice accumulation|using frequency instead of empirical odds due to error message related to predictions being length = 44 and trauma to no trauma matrix being length = 88 | additionally, error when using odds

y#plots ^