Archaeological\_linear\_mixed\_models

DC

2024-11-30

# 0. Prepare the work envrionment

## Install necessary packages

# Install necessary packages  
required\_packages <- c(  
 "tidyverse", "lme4", "MASS", "car", "lmerTest", "performance",   
 "splines", "brms", "elevatr", "raster", "ggplot2", "akima",   
 "fields", "sf", "sp", "RColorBrewer", "gstat", "broom.mixed"  
)  
  
# Install missing packages  
install\_missing <- required\_packages[!required\_packages %in% installed.packages()[, "Package"]]  
if (length(install\_missing) > 0) {  
 install.packages(install\_missing)  
}  
  
# Load libraries  
lapply(required\_packages, library, character.only = TRUE)

# 1. Import the dataset and prepare the data

ancient\_chi\_detail <- read\_csv("data/ancient\_chinese\_detail\_for\_mixed\_effects\_july.csv")  
ancient\_chi\_detail <- ancient\_chi\_detail %>%   
 mutate(time\_period = as.character(time\_period),  
 Subsistence = as.character(Subsistence))

## Assign the climatic data

coordinates\_ancient\_detail <- ancient\_chi\_detail %>%  
 dplyr::select(longitude, latitude)   
  
# Convert to a spatial points object  
points\_ancient\_detail <- SpatialPoints(coordinates\_ancient\_detail)

### Extract raster values for each point

min\_temp\_values\_ancient\_detail <- extract(Min\_temp, points\_ancient\_detail)  
max\_temp\_values\_ancient\_detail <- extract(Max\_temp, points\_ancient\_detail)  
min\_precip\_values\_ancient\_detail <- extract(Min\_precip, points\_ancient\_detail)  
max\_precip\_values\_ancient\_detail <- extract(Max\_precip, points\_ancient\_detail)

### Add the extracted data back to the dataset

ancient\_chi\_detail <- ancient\_chi\_detail %>%  
 mutate(Min\_temp = min\_temp\_values\_ancient\_detail,  
 Max\_temp = max\_temp\_values\_ancient\_detail,  
 Min\_precip = min\_precip\_values\_ancient\_detail,  
 Max\_precip = max\_precip\_values\_ancient\_detail)

## Seperate males and females

ancient\_chi\_male <- ancient\_chi\_detail %>%  
 filter(sex == '0')  
ancient\_chi\_female <- ancient\_chi\_detail %>%  
 filter(sex == '1')

# 2. Influences from the environment and time periods

## Remove rows with missing values in the columns used in the models

### For FXL

ancient\_chi\_male\_FXL <- ancient\_chi\_male %>%  
 filter(!is.na(FXL))  
ancient\_chi\_female\_FXL <- ancient\_chi\_female %>%  
 filter(!is.na(FXL))

### For TXL

ancient\_chi\_male\_TXL <- ancient\_chi\_male %>%  
 filter(!is.na(TXL))  
ancient\_chi\_female\_TXL <- ancient\_chi\_female %>%  
 filter(!is.na(TXL))

### For HXL

ancient\_chi\_male\_HXL <- ancient\_chi\_male %>%  
 filter(!is.na(ancient\_chi\_male$HXL))  
ancient\_chi\_female\_HXL <- ancient\_chi\_female %>%  
 filter(!is.na(ancient\_chi\_female$HXL))

### For RXL

ancient\_chi\_male\_RXL <- ancient\_chi\_male %>%  
 filter(!is.na(ancient\_chi\_male$RXL))  
ancient\_chi\_female\_RXL <- ancient\_chi\_female %>%  
 filter(!is.na(ancient\_chi\_female$RXL))

### For FHD

ancient\_chi\_male\_FHD <- ancient\_chi\_male %>%  
 filter(!is.na(FHD))  
ancient\_chi\_female\_FHD <- ancient\_chi\_female %>%  
 filter(!is.na(FHD))

## 2.1 Basic Linear regression

### Fit the full linear model using lm

#### For FXL

lm\_ancient\_male\_fxl\_full <- lm(data = ancient\_chi\_male\_FXL, FXL ~ altitude\_range + Min\_temp + Max\_temp + Min\_precip + Max\_precip + time\_period ) # using time\_period rather than subsistence to keep independence of each predictor  
lm\_ancient\_female\_fxl\_full <- lm(data = ancient\_chi\_female\_FXL, FXL ~ altitude\_range + Min\_temp + Max\_temp + Min\_precip + Max\_precip + time\_period )

#### For TXL

lm\_ancient\_male\_txl\_full <- lm(data = ancient\_chi\_male\_TXL, TXL ~ altitude\_range + Min\_temp + Max\_temp + Min\_precip + Max\_precip + time\_period ) # using time\_period rather than subsistence to keep independence of each predictor  
lm\_ancient\_female\_txl\_full <- lm(data = ancient\_chi\_female\_TXL, TXL ~ altitude\_range + Min\_temp + Max\_temp + Min\_precip + Max\_precip + time\_period )

#### For FHD

lm\_ancient\_male\_fhd\_full <- lm(data = ancient\_chi\_male\_FHD, FHD ~ altitude\_range+ Min\_temp + Max\_temp + Min\_precip + Max\_precip + time\_period ) # using time\_period rather than subsistence to keep independence of each predictor  
lm\_ancient\_female\_fhd\_full <- lm(data = ancient\_chi\_female\_FHD, FHD ~ altitude\_range + Min\_temp + Max\_temp + Min\_precip + Max\_precip + time\_period )

## 2.2 Mixed effects analyis

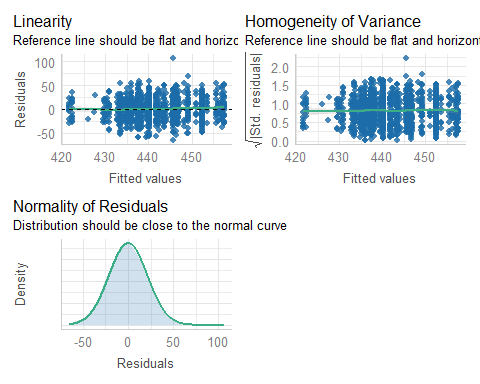
### Fit the full mixed effects model using lmer

#### For FXL

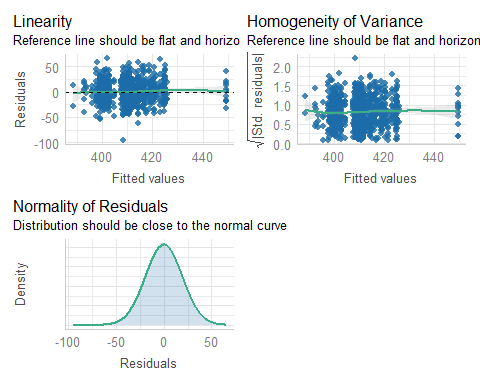
# Fit the model again with lmerTest loaded  
lme\_male\_fxl\_full1 <- lmer(FXL ~ altitude\_range + Min\_temp + Max\_temp + Min\_precip + Max\_precip + time\_period + (1 | site\_id), data = ancient\_chi\_male\_FXL)  
lme\_female\_fxl\_full1 <- lmer(FXL ~ altitude\_range + Min\_temp + Max\_temp + Min\_precip + Max\_precip + time\_period + (1 | site\_id), data = ancient\_chi\_female\_FXL)

#### Check the model

check\_model(lme\_male\_fxl\_full1, check = c("linearity", "homogeneity", "normality"))



check\_model(lme\_female\_fxl\_full1, check = c("linearity", "homogeneity", "normality"))

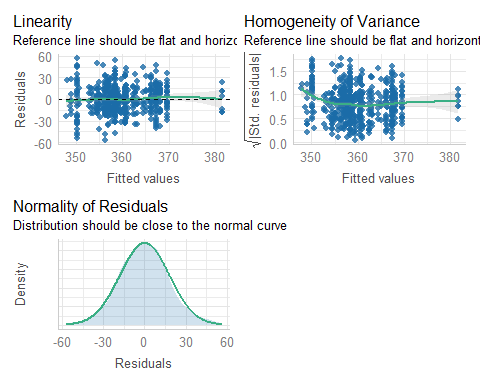


#### For TXL

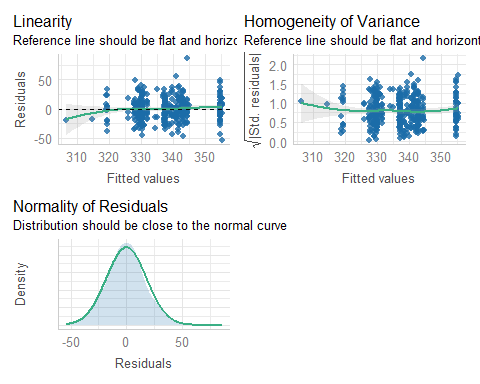
# Fit the model again with lmerTest loaded  
lme\_male\_txl\_full1 <- lmer(TXL ~ altitude\_range + Min\_temp + Max\_temp + Min\_precip + Max\_precip + time\_period + (1 | site\_id), data = ancient\_chi\_male\_TXL)  
lme\_female\_txl\_full1 <- lmer(TXL ~ altitude\_range + Min\_temp + Max\_temp + Min\_precip + Max\_precip + time\_period + (1 | site\_id), data = ancient\_chi\_female\_TXL)

#### Check the model

check\_model(lme\_male\_txl\_full1, check = c("linearity", "homogeneity", "normality"))



check\_model(lme\_female\_txl\_full1, check = c("linearity", "homogeneity", "normality"))

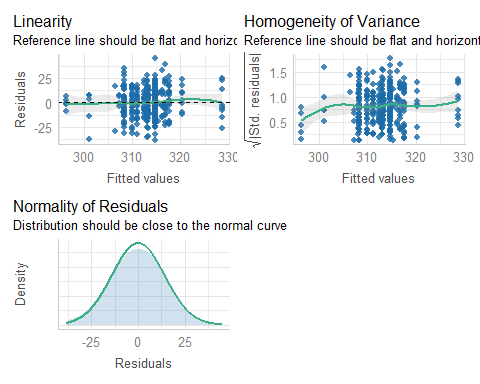


#### For HXL

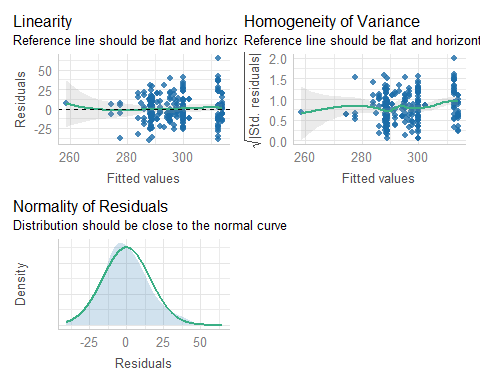
# Fit the model again with lmerTest loaded  
lme\_male\_hxl\_full1 <- lmer(HXL ~ altitude\_range + Min\_temp + Min\_precip + Max\_precip + time\_period + (1 | site\_id), data = ancient\_chi\_male\_HXL)  
lme\_female\_hxl\_full1 <- lmer(HXL ~ altitude\_range + Min\_temp + Min\_precip + Max\_precip + time\_period + (1 | site\_id), data = ancient\_chi\_female\_HXL)

#### Check the model

check\_model(lme\_male\_hxl\_full1, check = c("linearity", "homogeneity", "normality"))



check\_model(lme\_female\_hxl\_full1, check = c("linearity", "homogeneity", "normality"))

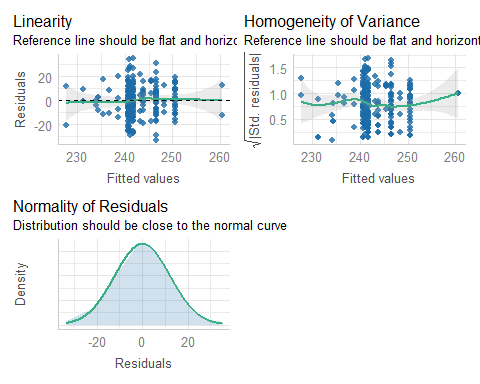


#### For RXL

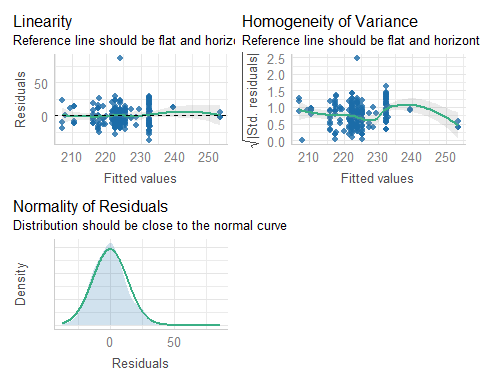
# Fit the model again with lmerTest loaded  
lme\_male\_rxl\_full1 <- lmer(RXL ~ altitude\_range + Min\_temp + Min\_precip + Max\_precip + time\_period + (1 | site\_id), data = ancient\_chi\_male\_RXL)  
lme\_female\_rxl\_full1 <- lmer(RXL ~ altitude\_range + Min\_temp + Min\_precip + Max\_precip + time\_period + (1 | site\_id), data = ancient\_chi\_female\_RXL)

#### Check the model

check\_model(lme\_male\_rxl\_full1, check = c("linearity", "homogeneity", "normality"))



check\_model(lme\_female\_rxl\_full1, check = c("linearity", "homogeneity", "normality"))

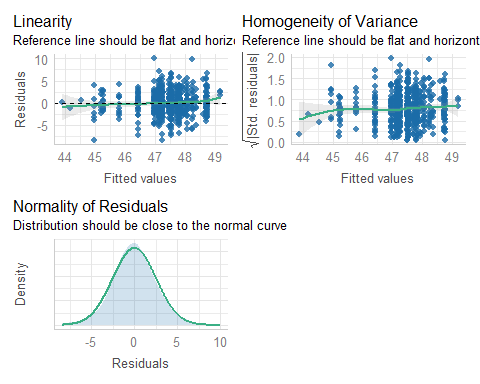


#### For FHD

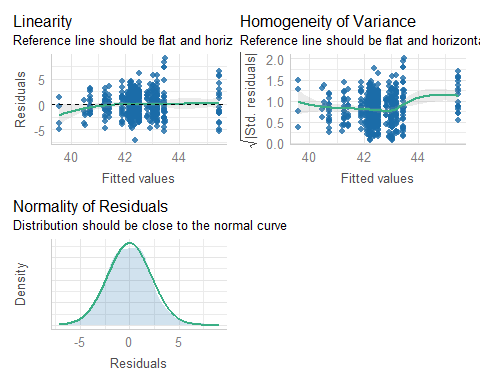
# Fit the model again with lmerTest loaded  
lme\_male\_fhd\_full1 <- lmer(FHD ~ altitude\_range + Min\_temp + Max\_temp + Min\_precip + Max\_precip + time\_period + (1 | site\_id), data = ancient\_chi\_male\_FHD)  
lme\_female\_fhd\_full1 <- lmer(FHD ~ altitude\_range + Min\_temp + Max\_temp + Min\_precip + Max\_precip + time\_period + (1 | site\_id), data = ancient\_chi\_female\_FHD)

#### Check the model

check\_model(lme\_male\_fhd\_full1, check = c("linearity", "homogeneity", "normality"))



check\_model(lme\_female\_fhd\_full1, check = c("linearity", "homogeneity", "normality"))



### Plot Coefficients for full models

#### FXL

# Tidy the model outputs and add model labels  
tidy\_male\_full\_fxl <- tidy(lme\_male\_fxl\_full1, effects = "fixed") %>%  
 mutate(model = "Male")  
tidy\_female\_full\_fxl <- tidy(lme\_female\_fxl\_full1, effects = "fixed") %>%  
 mutate(model = "Female")  
  
# Combine male and female outputs  
tidy\_combined\_full\_fxl <- bind\_rows(tidy\_male\_full\_fxl, tidy\_female\_full\_fxl) %>%  
 filter(term != "(Intercept)") %>% # Remove intercept  
 mutate(  
 sig = ifelse(p.value < 0.05, "Sig", "Not Sig"), # Add significance flag  
 ci\_low = estimate - 1.96 \* std.error, # Compute lower CI  
 ci\_high = estimate + 1.96 \* std.error # Compute upper CI  
 )  
  
# Reorder terms for plotting  
tidy\_combined\_full\_fxl$term <- factor(tidy\_combined\_full\_fxl$term, levels = unique(tidy\_combined\_full\_fxl$term))  
  
# Enhanced plot with confidence intervals  
plot\_full\_fxl\_coefficients <- ggplot(tidy\_combined\_full\_fxl, aes(x = estimate, y = term, fill = interaction(model, sig))) +  
 geom\_bar(stat = "identity", position = position\_dodge(width = 0.7), width = 0.6) +  
 geom\_errorbar(aes(xmin = ci\_low, xmax = ci\_high),   
 position = position\_dodge(0.7), width = 0.2) +  
 labs(  
 y = "Predictor Variables",   
 x = "Coefficient Estimate (with 95% Confidence Interval)",   
 title = "Fixed Effects Coefficient Estimates for Male and Female Models (FXL)"  
 ) +  
 # Custom fill scale to distinguish significance  
 scale\_fill\_manual(values = c(  
 "Male.Sig" = "blue",   
 "Female.Sig" = "red",   
 "Male.Not Sig" = "lightblue",   
 "Female.Not Sig" = "pink"  
 )) +  
 theme\_minimal() +  
 theme(  
 axis.text.y = element\_text(angle = 0, hjust = 1, size = 16), # Readable y-axis text  
 axis.text.x = element\_text(angle = 45, hjust = 1, size = 16), # Readable x-axis text  
 axis.title.y = element\_text(size = 16), # Enlarged y-axis title  
 axis.title.x = element\_text(size = 16), # Enlarged x-axis title  
 plot.title = element\_text(hjust = 0.5, size = 18), # Centered and enlarged title  
 legend.position = "bottom", # Legend at bottom  
 legend.text = element\_text(size = 16), # Enlarged legend text  
 legend.title = element\_text(size = 16) # Enlarged legend title  
 ) +  
 coord\_flip()

#### TXL

# Tidy the model outputs and add model labels  
tidy\_male\_full\_txl <- tidy(lme\_male\_txl\_full1, effects = "fixed") %>%  
 mutate(model = "Male")  
tidy\_female\_full\_txl <- tidy(lme\_female\_txl\_full1, effects = "fixed") %>%  
 mutate(model = "Female")  
  
# Combine male and female outputs  
tidy\_combined\_full\_txl <- bind\_rows(tidy\_male\_full\_txl, tidy\_female\_full\_txl) %>%  
 filter(term != "(Intercept)") %>% # Remove intercept  
 mutate(  
 sig = ifelse(p.value < 0.05, "Sig", "Not Sig"), # Add significance flag  
 ci\_low = estimate - 1.96 \* std.error, # Compute lower CI  
 ci\_high = estimate + 1.96 \* std.error # Compute upper CI  
 )  
  
# Reorder terms for plotting  
tidy\_combined\_full\_txl$term <- factor(tidy\_combined\_full\_txl$term, levels = unique(tidy\_combined\_full\_txl$term))  
  
# Create the plot  
plot\_full\_txl\_coefficients <- ggplot(tidy\_combined\_full\_txl, aes(y = term, x = estimate, fill = interaction(model, sig))) +  
 geom\_bar(stat = "identity", position = position\_dodge(width = 0.7), width = 0.6) +  
 geom\_errorbar(aes(xmin = ci\_low, xmax = ci\_high),   
 position = position\_dodge(0.7), width = 0.2) +  
 labs(  
 y = "Predictor Variables",   
 x = "Coefficient Estimate (with 95% Confidence Interval)",   
 title = "Fixed Effects Coefficient Estimates for Male and Female Models (TXL)"  
 ) +  
 # Custom fill scale for significance and model interaction  
 scale\_fill\_manual(values = c(  
 "Male.Sig" = "blue",   
 "Female.Sig" = "red",   
 "Male.Not Sig" = "lightblue",   
 "Female.Not Sig" = "pink"  
 )) +  
 theme\_minimal() +  
 theme(  
 axis.text.y = element\_text(angle = 0, hjust = 1, size = 16), # Readable y-axis text  
 axis.text.x = element\_text(angle = 45, hjust = 1, size = 16), # Readable x-axis text  
 axis.title.y = element\_text(size = 16), # Enlarged y-axis title  
 axis.title.x = element\_text(size = 16), # Enlarged x-axis title  
 plot.title = element\_text(hjust = 0.5, size = 18), # Centered and enlarged title  
 legend.position = "bottom", # Legend at the bottom  
 legend.text = element\_text(size = 16), # Enlarged legend text  
 legend.title = element\_text(size = 16) # Enlarged legend title  
 ) +  
 coord\_flip()

# Tidy the model outputs and add model labels  
tidy\_male\_full\_fhd <- tidy(lme\_male\_fhd\_full1, effects = "fixed") %>%  
 mutate(model = "Male")  
tidy\_female\_full\_fhd <- tidy(lme\_female\_fhd\_full1, effects = "fixed") %>%  
 mutate(model = "Female")  
  
# Combine male and female outputs  
tidy\_combined\_full\_fhd <- bind\_rows(tidy\_male\_full\_fhd, tidy\_female\_full\_fhd) %>%  
 filter(term != "(Intercept)") %>% # Remove intercept  
 mutate(  
 sig = ifelse(p.value < 0.05, "Sig", "Not Sig"), # Add significance flag  
 ci\_low = estimate - 1.96 \* std.error, # Compute lower CI  
 ci\_high = estimate + 1.96 \* std.error # Compute upper CI  
 )  
  
# Reorder terms for plotting  
tidy\_combined\_full\_fhd$term <- factor(tidy\_combined\_full\_fhd$term, levels = unique(tidy\_combined\_full\_fhd$term))  
  
# Create the plot  
plot\_full\_fhd\_coefficients <- ggplot(tidy\_combined\_full\_fhd, aes(y = term, x = estimate, fill = interaction(model, sig))) +  
 geom\_bar(stat = "identity", position = position\_dodge(width = 0.7), width = 0.6) +  
 geom\_errorbar(aes(xmin = ci\_low, xmax = ci\_high),   
 position = position\_dodge(0.7), width = 0.2) +  
 labs(  
 y = "Predictor Variables",   
 x = "Coefficient Estimate (with 95% Confidence Interval)",   
 title = "Fixed Effects Coefficient Estimates for Male and Female Models (FHD)"  
 ) +  
 scale\_fill\_manual(values = c(  
 "Male.Sig" = "blue",   
 "Female.Sig" = "red",   
 "Male.Not Sig" = "lightblue",   
 "Female.Not Sig" = "pink"  
 )) +  
 theme\_minimal() +  
 theme(  
 axis.text.y = element\_text(angle = 0, hjust = 1, size = 16),  
 axis.text.x = element\_text(angle = 45, hjust = 1, size = 16),  
 axis.title.y = element\_text(size = 16),  
 axis.title.x = element\_text(size = 16),  
 plot.title = element\_text(hjust = 0.5, size = 18),  
 legend.position = "bottom",  
 legend.text = element\_text(size = 16),  
 legend.title = element\_text(size = 16)  
 ) +  
 coord\_flip()