synthesis code

Patrick Das

2025-07-07

# Data cleaning and transformation

## Load packages

library(tidyr)  
library(readr)  
library(dplyr)  
library(readr)  
library(ggplot2)  
library(ggrepel)  
library(nnet)  
library(lmtest)  
library(nortest)  
library(spdep)  
library(lme4)  
library(lmerTest)   
library(MuMIn)  
  
data <- read\_tsv("synthesis aggregrate data.txt")  
  
data <- data[, colSums(is.na(data)) == 0] #dropping columns with NA values  
  
data[data == "N/A"] <- NA #rewriting "NA" values in text to reflect actual NA  
  
data <- data %>% drop\_na() #dropping observations with NA values

## Data preparation

### Converting variables to factors

data <- data %>%  
 mutate(across(where(is.character), as.factor)) #converting chr to factors

### Cleaning column names

# Create clean, consistent column names  
clean\_names <- names(data)  
  
# Replace spaces and punctuation with underscores  
clean\_names <- gsub(" ", "\_", clean\_names) # spaces → \_  
clean\_names <- gsub("\\?", "", clean\_names) # remove question marks  
clean\_names <- gsub("\\(", "", clean\_names) # remove (  
clean\_names <- gsub("\\)", "", clean\_names) # remove )  
clean\_names <- gsub("\\/", "\_", clean\_names) # slashes → \_  
clean\_names <- gsub("\\+", "plus", clean\_names) # plus signs → 'plus'  
clean\_names <- gsub("-", "\_", clean\_names) # hyphens → \_  
clean\_names <- gsub("\_\_+", "\_", clean\_names) # collapse double underscores  
clean\_names <- gsub("[^[:alnum:]\_]", "", clean\_names) # remove anything not alphanumeric or underscore  
clean\_names <- tolower(clean\_names) # make all lowercase (optional)  
  
# Apply new names to the dataframe  
names(data) <- clean\_names  
  
# Applying custom mapping  
names(data)[names(data) == "granular\_case\_marking\_alignment\_type"] <- "granular\_alignment"  
names(data)[names(data) == "case\_marking\_complexity\_distinctions\_plus\_optionality"] <- "case\_marking\_complexity"  
names(data)[names(data) == "spoken\_as\_an\_l2"] <- "L2"  
names(data)[names(data) == "speaker\_population"] <- "population"  
names(data)[names(data) == "agriculture\_intensity\_based\_on\_ea028"] <- "agricultural\_intensity"  
names(data)[names(data) == "political\_organization\_ea033"] <- "political\_organization"  
names(data)[names(data) == "altitude\_sampled\_raster\_value\_from\_dem"] <- "altitude"  
names(data)[names(data) == "subfamily\_node\_below\_top\_level"] <- "subfamily"  
names(data)[names(data) == "language\_family\_top\_level\_node"] <- "family"

### Creating/transforming variables

#### Binarizing agricultural intensity

data$agriculture\_binary <- ifelse(  
 data$agricultural\_intensity == "Intensive/irrigated", 1, 0  
)  
table(data$agriculture\_binary)

##   
## 0 1   
## 26 6

#### Binarizing political organization

data$political\_organization\_binary <- ifelse(  
 data$political\_organization == "State", 1, 0  
 )  
table(data$political\_organization\_binary)

##   
## 0 1   
## 27 5

#### Binarizing hill/valley (needed for ecological mode)

# Create hill\_binary variable (1 = Hill, 0 = Valley or Split)  
data$hill\_binary <- ifelse(data$hill\_valley == "Hill", 1, 0)  
  
table(data$hill\_valley, data$hill\_binary)

##   
## 0 1  
## Hill 0 18  
## Split 8 0  
## Valley 6 0

#### Make a new column for whether a language has verbal indexing

data$v\_indexing <- ifelse(data$indexing %in% c("S", "O", "S,A", "S,A,O"), 1, 0)  
write.csv(data, "cleaned\_data\_28\_05\_25.csv")

#### Scaling variables

Scaling population

#let's scale populaiton  
data$scaled\_population <- scale(data$population)  
# Scaling geographic predictors  
data$scaled\_altitude <- scale(data$altitude)[, 1]  
data$scaled\_stdev\_slope <- scale(data$stdev\_of\_slope)[, 1]

Scaling complexity scores

#First, individual measures of complexity are scaled  
data$scaled\_case\_marking\_complexity <- scale(data$case\_marking\_complexity)  
data$scaled\_form\_complexity <- scale(data$form\_complexity)  
data$scaled\_cell\_complexity <- scale(data$cell\_complexity)

# Initial Analysis (with Limbu)

## Hill/Valley classification

Predicting the complexity of a language system based on hill/valley classification

# Cell complexity model  
model\_cell <- lmer(scaled\_cell\_complexity ~ hill\_valley + (1 | subfamily), data = data)  
summary(model\_cell)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_cell\_complexity ~ hill\_valley + (1 | subfamily)  
## Data: data  
##   
## REML criterion at convergence: 89.2  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -0.80673 -0.63454 -0.43850 0.04919 3.14000   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.05934 0.2436   
## Residual 0.96029 0.9799   
## Number of obs: 32, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) 0.1439 0.2494 9.7905 0.577 0.577  
## hill\_valleySplit -0.5497 0.4205 25.3298 -1.307 0.203  
## hill\_valleyValley -0.1003 0.4712 28.7941 -0.213 0.833  
##   
## Correlation of Fixed Effects:  
## (Intr) hll\_vS  
## hll\_vllySpl -0.489   
## hll\_vllyVll -0.464 0.277

# Form complexity model  
model\_form <- lmer(scaled\_form\_complexity ~ hill\_valley + (1 | subfamily), data = data)  
summary(model\_form)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_form\_complexity ~ hill\_valley + (1 | subfamily)  
## Data: data  
##   
## REML criterion at convergence: 90  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -0.7793 -0.5157 -0.4157 -0.1195 3.3760   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.06547 0.2559   
## Residual 0.98414 0.9920   
## Number of obs: 32, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) 0.04967 0.25381 12.26296 0.196 0.848  
## hill\_valleySplit -0.32898 0.42592 26.30197 -0.772 0.447  
## hill\_valleyValley 0.05613 0.47770 28.85622 0.117 0.907  
##   
## Correlation of Fixed Effects:  
## (Intr) hll\_vS  
## hll\_vllySpl -0.485   
## hll\_vllyVll -0.462 0.277

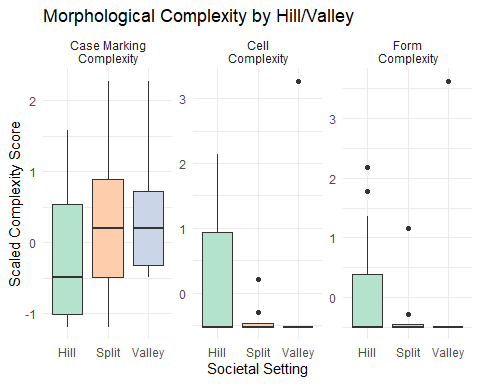
# Case marking complexity model  
model\_case <- lmer(scaled\_case\_marking\_complexity ~ hill\_valley + (1 | subfamily), data = data)  
summary(model\_case)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_case\_marking\_complexity ~ hill\_valley + (1 | subfamily)  
## Data: data  
##   
## REML criterion at convergence: 88  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.4678 -0.7817 -0.2529 0.5476 2.0432   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.0312 0.1766   
## Residual 0.9386 0.9688   
## Number of obs: 32, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.2635 0.2386 12.0910 -1.104 0.291  
## hill\_valleySplit 0.5418 0.4140 26.5706 1.309 0.202  
## hill\_valleyValley 0.7119 0.4619 28.8463 1.541 0.134  
##   
## Correlation of Fixed Effects:  
## (Intr) hll\_vS  
## hll\_vllySpl -0.516   
## hll\_vllyVll -0.479 0.277

# Visualizing data, identifying and removing outliers, and testing assumptions

## Visualizing complexity against hill/valley via box-plots

# Reshape the data  
data\_long <- data %>%  
 select(hill\_valley, scaled\_cell\_complexity, scaled\_form\_complexity, scaled\_case\_marking\_complexity) %>%  
 pivot\_longer(  
 cols = starts\_with("scaled\_"),  
 names\_to = "complexity\_type",  
 values\_to = "score"  
 ) %>%  
 mutate(complexity\_type = case\_when(  
 complexity\_type == "scaled\_cell\_complexity" ~ "Cell Complexity",  
 complexity\_type == "scaled\_form\_complexity" ~ "Form Complexity",  
 complexity\_type == "scaled\_case\_marking\_complexity" ~ "Case Marking Complexity"  
 ))  
ggplot(data\_long, aes(x = hill\_valley, y = score, fill = hill\_valley)) +  
 geom\_boxplot() +  
 facet\_wrap(~ complexity\_type, scales = "free\_y",   
 labeller = label\_wrap\_gen(width = 15)) +  
 labs(  
 title = "Morphological Complexity by Hill/Valley",  
 x = "Societal Setting", y = "Scaled Complexity Score"  
 ) +  
 scale\_fill\_brewer(palette = "Pastel2") +  
 theme\_minimal() +  
 theme(legend.position = "none")



## sensitivity testing

Identifying outliers and influential values

form\_cooks <- cooks.distance(model\_form)  
cell\_cooks <- cooks.distance(model\_cell)  
case\_cooks <- cooks.distance(model\_case)  
  
# Set threshold  
threshold <- 4 / nrow(data)  
  
# Identify outliers  
form\_outlier <- which(form\_cooks > threshold)  
cell\_outlier <- which(cell\_cooks > threshold)  
case\_outlier <- which(case\_cooks >threshold)  
  
  
#and most influential  
which.max(form\_cooks)

## 23   
## 23

which.max(cell\_cooks)

## 23   
## 23

which.max(case\_cooks)

## 2   
## 2

# For Limbu  
limbu\_form <- form\_cooks[data$language == "Limbu"]  
limbu\_cell <- cell\_cooks[data$language == "Limbu"]  
  
# For Bori-Karko  
bori\_case <- case\_cooks[data$language == "Bori-Karko"]  
  
#printing out most influential observations  
# For form complexity  
data[which.max(form\_cooks), ]

## # A tibble: 1 × 51  
## language pid glottocode mother\_node subfamily family reference alignment  
## <fct> <fct> <fct> <fct> <fct> <fct> <fct> <fct>   
## 1 Limbu STKI01 limb1266 Kiranti Himalayish Sino-Ti… Driem, G… ERG-ABS   
## # ℹ 43 more variables: granular\_alignment <fct>, optional\_a <dbl>,  
## # optional\_o <dbl>, case\_marking\_complexity <dbl>, indexing <fct>,  
## # heirarchical <fct>, inverse\_marking <dbl>, cell\_complexity <dbl>,  
## # form\_complexity <dbl>, verbal\_complexity\_sum <dbl>, total\_complexity <dbl>,  
## # latitude <dbl>, longitude <dbl>, L2 <dbl>, population <dbl>,  
## # speaker\_population\_comments <fct>, agricultural\_intensity <fct>,  
## # agriculture\_comments <fct>, political\_organization <fct>, …

limbu\_form

## 23   
## 1.202295

# For cell complexity  
data[which.max(cell\_cooks), ]

## # A tibble: 1 × 51  
## language pid glottocode mother\_node subfamily family reference alignment  
## <fct> <fct> <fct> <fct> <fct> <fct> <fct> <fct>   
## 1 Limbu STKI01 limb1266 Kiranti Himalayish Sino-Ti… Driem, G… ERG-ABS   
## # ℹ 43 more variables: granular\_alignment <fct>, optional\_a <dbl>,  
## # optional\_o <dbl>, case\_marking\_complexity <dbl>, indexing <fct>,  
## # heirarchical <fct>, inverse\_marking <dbl>, cell\_complexity <dbl>,  
## # form\_complexity <dbl>, verbal\_complexity\_sum <dbl>, total\_complexity <dbl>,  
## # latitude <dbl>, longitude <dbl>, L2 <dbl>, population <dbl>,  
## # speaker\_population\_comments <fct>, agricultural\_intensity <fct>,  
## # agriculture\_comments <fct>, political\_organization <fct>, …

limbu\_cell

## 23   
## 1.02319

# For case complexity  
data[which.max(case\_cooks), ]

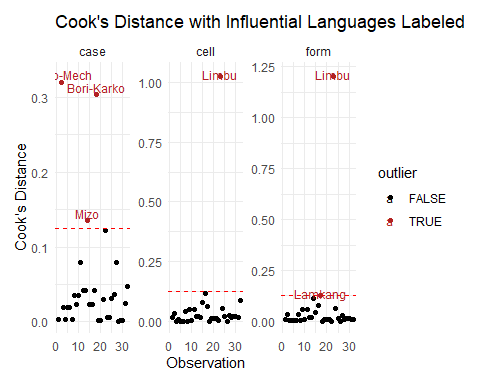
## # A tibble: 1 × 51  
## language pid glottocode mother\_node subfamily family reference alignment  
## <fct> <fct> <fct> <fct> <fct> <fct> <fct> <fct>   
## 1 Bodo-Mech STBG01 bodo1269 Bodo-Mech-Ka… Sal (Bra… Sino-… Brahma 2… NOM-ACC   
## # ℹ 43 more variables: granular\_alignment <fct>, optional\_a <dbl>,  
## # optional\_o <dbl>, case\_marking\_complexity <dbl>, indexing <fct>,  
## # heirarchical <fct>, inverse\_marking <dbl>, cell\_complexity <dbl>,  
## # form\_complexity <dbl>, verbal\_complexity\_sum <dbl>, total\_complexity <dbl>,  
## # latitude <dbl>, longitude <dbl>, L2 <dbl>, population <dbl>,  
## # speaker\_population\_comments <fct>, agricultural\_intensity <fct>,  
## # agriculture\_comments <fct>, political\_organization <fct>, …

bori\_case

## 18   
## 0.3037258

## Visualizing outliers

# Creating tidy Cook's data with language preserved  
cook\_data <- data %>%  
 mutate(  
 obs = row\_number(),  
 form = form\_cooks,  
 cell = cell\_cooks,  
 case = case\_cooks  
 ) %>%  
 select(obs, language, form, cell, case)  
  
# Pivoting to long format for ggplot  
cook\_long <- cook\_data %>%  
 pivot\_longer(cols = c(form, cell, case),  
 names\_to = "model", values\_to = "cooks") %>%  
 mutate(  
 outlier = cooks > threshold,  
 label = ifelse(outlier, as.character(language), NA)  
 )  
  
# Plotting with labeled outliers  
ggplot(cook\_long, aes(x = obs, y = cooks, color = outlier, label = label)) +  
 geom\_point() +  
 geom\_hline(yintercept = threshold, linetype = "dashed", color = "red") +  
 geom\_text(nudge\_y = 0.01, check\_overlap = TRUE, size = 3.2) +  
 facet\_wrap(~ model, scales = "free\_y") +  
 theme\_minimal() +  
 labs(title = "Cook's Distance with Influential Languages Labeled",  
 x = "Observation", y = "Cook's Distance") +  
 scale\_color\_manual(values = c("black", "firebrick"))+  
 theme(plot.margin = margin(10, 20, 10, 10)) # top, right, bottom, left (in pts)



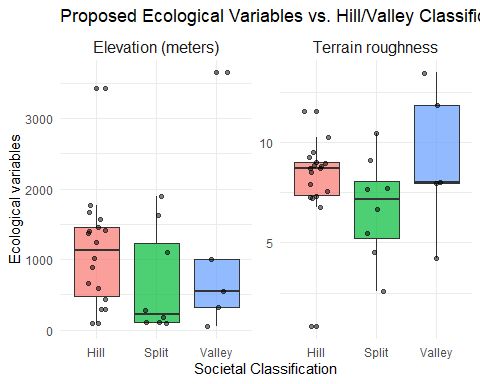
## Creating a new dataset without Limbu and Bodo-Mech

data\_sensitivity <- data[data$language != "Limbu", ]

## Visualizing variables

### Visualizing ecological variables against hill/valley classification

# Create long-format data for both terrain variables  
terrain\_data <- data\_sensitivity %>%  
 select(hill\_valley, altitude, stdev\_of\_slope) %>%  
 pivot\_longer(  
 cols = c(altitude, stdev\_of\_slope),  
 names\_to = "terrain\_variable",  
 values\_to = "value"  
 ) %>%  
 mutate(terrain\_variable = case\_when(  
 terrain\_variable == "altitude" ~ "Elevation (meters)",  
 terrain\_variable == "stdev\_of\_slope" ~ "Terrain roughness"  
 ))  
  
# Create faceted boxplot  
ggplot(terrain\_data, aes(x = hill\_valley, y = value, fill = hill\_valley)) +  
 geom\_boxplot(alpha = 0.7) +  
 geom\_jitter(width = 0.2, alpha = 0.5) +  
 facet\_wrap(~ terrain\_variable, scales = "free\_y") +  
 labs(  
 title = "Proposed Ecological Variables vs. Hill/Valley Classification",  
 x = "Societal Classification",   
 y = "Ecological variables",  
 fill = "Classification"  
 ) +  
 theme\_minimal() +  
 theme(  
 legend.position = "none",  
 strip.text = element\_text(size = 12)  
 )



# Analysis (after removing Limbu)

## Predicting complexity based on hill/valley classification

# Cell complexity model  
model\_cell\_sensitivity <- lmer(scaled\_cell\_complexity ~ hill\_valley + (1 | subfamily),  
 data = data\_sensitivity)  
  
# Form complexity model  
model\_form\_sensitivity <- lmer(scaled\_form\_complexity ~ hill\_valley + (1 | subfamily),  
 data = data\_sensitivity)  
  
# Case complexity model  
model\_case\_sensitivity <- lmer(scaled\_case\_marking\_complexity ~ hill\_valley + (1 | subfamily),  
 data = data\_sensitivity)  
  
# View summaries  
summary(model\_cell\_sensitivity)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_cell\_complexity ~ hill\_valley + (1 | subfamily)  
## Data: data\_sensitivity  
##   
## REML criterion at convergence: 71.4  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.3075 -0.6302 -0.1934 0.2740 2.3521   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.09666 0.3109   
## Residual 0.52724 0.7261   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)   
## (Intercept) 0.0982 0.2068 12.9859 0.475 0.6428   
## hill\_valleySplit -0.5828 0.3153 25.1692 -1.848 0.0763 .  
## hill\_valleyValley -0.7304 0.3773 26.1974 -1.936 0.0637 .  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Correlation of Fixed Effects:  
## (Intr) hll\_vS  
## hll\_vllySpl -0.408   
## hll\_vllyVll -0.377 0.270

summary(model\_form\_sensitivity)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_form\_complexity ~ hill\_valley + (1 | subfamily)  
## Data: data\_sensitivity  
##   
## REML criterion at convergence: 68.5  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.2900 -0.5587 -0.3737 0.1949 2.5994   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.08914 0.2986   
## Residual 0.47459 0.6889   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)   
## (Intercept) -0.01019 0.19691 15.82250 -0.052 0.9594   
## hill\_valleySplit -0.37110 0.29926 26.03480 -1.240 0.2260   
## hill\_valleyValley -0.63126 0.35810 26.76893 -1.763 0.0894 .  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Correlation of Fixed Effects:  
## (Intr) hll\_vS  
## hll\_vllySpl -0.406   
## hll\_vllyVll -0.375 0.270

summary(model\_case\_sensitivity)

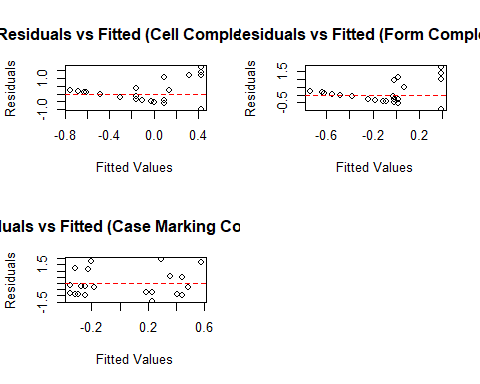
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_case\_marking\_complexity ~ hill\_valley + (1 | subfamily)  
## Data: data\_sensitivity  
##   
## REML criterion at convergence: 85.9  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.4424 -0.7961 -0.2582 0.5367 2.0089   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.03327 0.1824   
## Residual 0.96881 0.9843   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.2622 0.2428 11.6933 -1.080 0.302  
## hill\_valleySplit 0.5423 0.4207 25.6663 1.289 0.209  
## hill\_valleyValley 0.7592 0.5009 26.1333 1.516 0.142  
##   
## Correlation of Fixed Effects:  
## (Intr) hll\_vS  
## hll\_vllySpl -0.515   
## hll\_vllyVll -0.442 0.262

No significant effect (but directionality changed for Cell/Form models)

## Testing assumptions

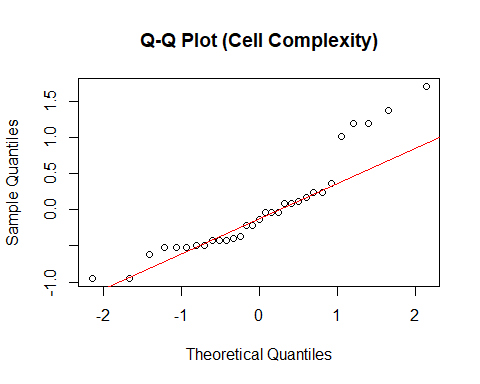
### Plotting residuals

plot\_residuals <- function(model, title) {  
 # Create plot  
 plot(fitted(model), residuals(model),  
 xlab = "Fitted Values", ylab = "Residuals",  
 main = paste("Residuals vs Fitted (", title, ")", sep=""))  
 # Add reference line  
 abline(h = 0, col = "red", lty = 2)  
}  
par(mfrow = c(2, 2)) # Set up a 2x2 plotting area  
  
# Plot each model  
plot\_residuals(model\_cell\_sensitivity, "Cell Complexity")  
plot\_residuals(model\_form\_sensitivity, "Form Complexity")  
plot\_residuals(model\_case\_sensitivity, "Case Marking Complexity")



### Q-Q line and Shapiro wilk test

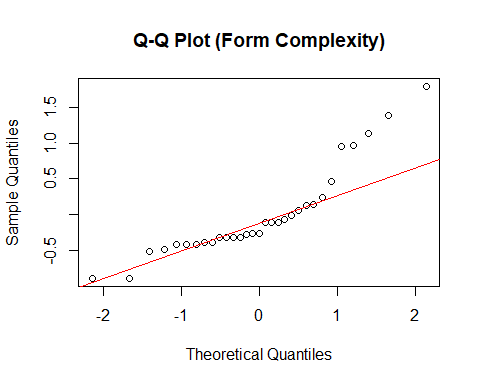
# Q-Q plot cell  
qqnorm(resid(model\_cell\_sensitivity), main = "Q-Q Plot (Cell Complexity)")  
qqline(resid(model\_cell\_sensitivity), col = "red")



# Shapiro-Wilk test  
shapiro.test(resid(model\_cell\_sensitivity))

##   
## Shapiro-Wilk normality test  
##   
## data: resid(model\_cell\_sensitivity)  
## W = 0.88631, p-value = 0.003338

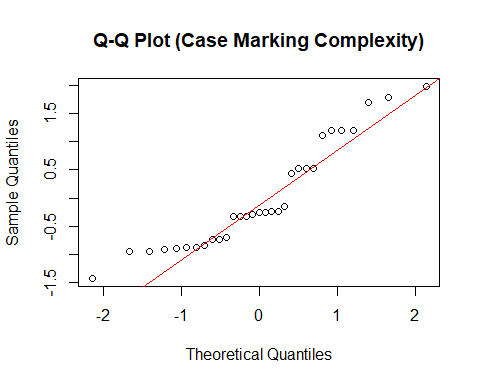
# Q-Q plot  
qqnorm(resid(model\_form\_sensitivity), main = "Q-Q Plot (Form Complexity)")  
qqline(resid(model\_form\_sensitivity), col = "red")



# Shapiro-Wilk test  
shapiro.test(resid(model\_form\_sensitivity))

##   
## Shapiro-Wilk normality test  
##   
## data: resid(model\_form\_sensitivity)  
## W = 0.85409, p-value = 0.0006199

# Q-Q plot  
qqnorm(resid(model\_case\_sensitivity), main = "Q-Q Plot (Case Marking Complexity)")  
qqline(resid(model\_case\_sensitivity), col = "red")

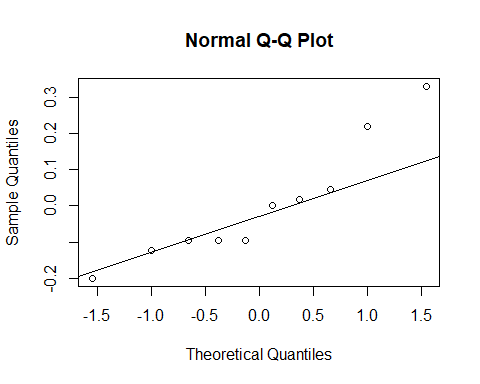


# Shapiro-Wilk test  
shapiro.test(resid(model\_case\_sensitivity))

##   
## Shapiro-Wilk normality test  
##   
## data: resid(model\_case\_sensitivity)  
## W = 0.90231, p-value = 0.008236

### Normality of random effects

# Extract random effects  
ranef\_subfam <- ranef(model\_cell\_sensitivity)$subfamily  
  
# Check normality of random effects  
qqnorm(ranef\_subfam[,1])  
qqline(ranef\_subfam[,1])



shapiro.test(ranef\_subfam[,1])

##   
## Shapiro-Wilk normality test  
##   
## data: ranef\_subfam[, 1]  
## W = 0.89186, p-value = 0.1779

## Predicting Hill/Valley classification from ecological variables

# Binarizing hill/valley classification: Hill = 1, others = 0  
data\_sensitivity$hill\_binary <- ifelse(data\_sensitivity$hill\_valley == "Hill", 1, 0)  
  
# Fitting logistic regression model  
hill\_ecology\_model <- glm(  
 hill\_binary ~ scaled\_altitude + scaled\_stdev\_slope,  
 family = binomial,  
 data = data\_sensitivity  
)  
  
hill\_altitude <- glm(hill\_binary ~ scaled\_altitude, family = binomial, data = data\_sensitivity)  
hill\_slope <- glm(hill\_binary ~ scaled\_stdev\_slope, family = binomial, data = data\_sensitivity)  
  
# Summarizing model output  
summary(hill\_ecology\_model)

##   
## Call:  
## glm(formula = hill\_binary ~ scaled\_altitude + scaled\_stdev\_slope,   
## family = binomial, data = data\_sensitivity)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) 0.34065 0.36940 0.922 0.356  
## scaled\_altitude 0.26203 0.45092 0.581 0.561  
## scaled\_stdev\_slope 0.07631 0.42190 0.181 0.856  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 42.165 on 30 degrees of freedom  
## Residual deviance: 41.516 on 28 degrees of freedom  
## AIC: 47.516  
##   
## Number of Fisher Scoring iterations: 4

summary(hill\_altitude)

##   
## Call:  
## glm(formula = hill\_binary ~ scaled\_altitude, family = binomial,   
## data = data\_sensitivity)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) 0.3414 0.3693 0.924 0.355  
## scaled\_altitude 0.3011 0.3964 0.760 0.448  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 42.165 on 30 degrees of freedom  
## Residual deviance: 41.548 on 29 degrees of freedom  
## AIC: 45.548  
##   
## Number of Fisher Scoring iterations: 4

summary(hill\_slope)

##   
## Call:  
## glm(formula = hill\_binary ~ scaled\_stdev\_slope, family = binomial,   
## data = data\_sensitivity)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) 0.3287 0.3659 0.898 0.369  
## scaled\_stdev\_slope 0.1999 0.3676 0.544 0.587  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 42.165 on 30 degrees of freedom  
## Residual deviance: 41.866 on 29 degrees of freedom  
## AIC: 45.866  
##   
## Number of Fisher Scoring iterations: 4

## Analysing complexity as predicted by sociocultural variables

### Complexity as predicted by L2\_status

# 1. L2 STATUS EFFECTS  
# Cell complexity model  
cell\_L2\_mixed <- lmer(scaled\_cell\_complexity ~ L2 + (1|subfamily), data = data\_sensitivity)  
  
# Form complexity model  
form\_L2\_mixed <- lmer(scaled\_form\_complexity ~ L2 + (1|subfamily), data = data\_sensitivity)  
  
# Case marking complexity model  
case\_L2\_mixed <- lmer(scaled\_case\_marking\_complexity ~ L2 + (1|subfamily), data = data\_sensitivity)  
  
# View L2 model summaries  
summary(cell\_L2\_mixed)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_cell\_complexity ~ L2 + (1 | subfamily)  
## Data: data\_sensitivity  
##   
## REML criterion at convergence: 71.9  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.1590 -0.7212 -0.0162 0.1134 2.4639   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.05746 0.2397   
## Residual 0.53802 0.7335   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)   
## (Intercept) 0.1095 0.1975 13.2996 0.554 0.5886   
## L2 -0.6175 0.2736 28.2245 -2.257 0.0319 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## L2 -0.553

summary(form\_L2\_mixed)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_form\_complexity ~ L2 + (1 | subfamily)  
## Data: data\_sensitivity  
##   
## REML criterion at convergence: 69.3  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.15604 -0.57483 -0.09536 0.00469 2.69230   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.06564 0.2562   
## Residual 0.48478 0.6963   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.008344 0.192554 16.303056 -0.043 0.966  
## L2 -0.433898 0.260986 28.401367 -1.663 0.107  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## L2 -0.535

summary(case\_L2\_mixed)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_case\_marking\_complexity ~ L2 + (1 | subfamily)  
## Data: data\_sensitivity  
##   
## REML criterion at convergence: 87.5  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.5013 -0.7873 -0.2984 0.5444 1.9849   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.0633 0.2516   
## Residual 0.9434 0.9713   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.2331 0.2508 8.5407 -0.929 0.378  
## L2 0.5560 0.3595 27.8088 1.546 0.133  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## L2 -0.582

#### If Limbu is added back in

# 1. L2 STATUS EFFECTS  
# Cell complexity model  
cell\_L2\_mixed\_orig <- lmer(scaled\_cell\_complexity ~ L2 + (1|subfamily), data = data)  
  
# Form complexity model  
form\_L2\_mixed\_orig <- lmer(scaled\_form\_complexity ~ L2 + (1|subfamily), data = data)  
  
# Case marking complexity model  
case\_L2\_mixed\_orig <- lmer(scaled\_case\_marking\_complexity ~ L2 + (1|subfamily), data = data)  
  
# View L2 model summaries  
summary(cell\_L2\_mixed\_orig)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_cell\_complexity ~ L2 + (1 | subfamily)  
## Data: data  
##   
## REML criterion at convergence: 90.6  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -0.7757 -0.6000 -0.3017 -0.1310 3.4175   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.03633 0.1906   
## Residual 0.97034 0.9851   
## Number of obs: 32, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) 0.1455 0.2453 9.7410 0.593 0.567  
## L2 -0.3425 0.3542 28.5300 -0.967 0.342  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## L2 -0.625

summary(form\_L2\_mixed\_orig)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_form\_complexity ~ L2 + (1 | subfamily)  
## Data: data  
##   
## REML criterion at convergence: 91.2  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -0.7694 -0.5046 -0.3301 -0.2183 3.5666   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.07714 0.2777   
## Residual 0.96180 0.9807   
## Number of obs: 32, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) 0.03861 0.25685 11.92134 0.150 0.883  
## L2 -0.13010 0.35549 28.52570 -0.366 0.717  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## L2 -0.592

summary(case\_L2\_mixed\_orig)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_case\_marking\_complexity ~ L2 + (1 | subfamily)  
## Data: data  
##   
## REML criterion at convergence: 89.4  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.5134 -0.7906 -0.2989 0.5284 2.0290   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.04988 0.2233   
## Residual 0.92038 0.9594   
## Number of obs: 32, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.2330 0.2440 9.4259 -0.955 0.363  
## L2 0.5404 0.3461 28.2402 1.561 0.130  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## L2 -0.611

#### Is effect of L2 on cell\_complexity robust to dropping languages?

# Create a dataframe to store results  
drop\_one\_results <- data.frame(  
 dropped\_language = character(),  
 estimate = numeric(),  
 std\_error = numeric(),  
 p\_value = numeric(),  
 significant = logical(),  
 stringsAsFactors = FALSE  
)  
  
# Loop through each language in data\_sensitivity  
for (lang in unique(data\_sensitivity$language)) {  
 # Create dataset with one additional language dropped  
 temp\_data <- data\_sensitivity[data\_sensitivity$language != lang, ]  
   
 # Fit the model  
 temp\_model <- lmer(scaled\_cell\_complexity ~ L2 + (1|subfamily), data = temp\_data)  
   
 # Extract model summary  
 model\_summary <- summary(temp\_model)  
   
 # Get L2 coefficient (second row in fixed effects table)  
 l2\_coef <- model\_summary$coefficients[2, "Estimate"]  
 l2\_se <- model\_summary$coefficients[2, "Std. Error"]  
 l2\_p <- model\_summary$coefficients[2, "Pr(>|t|)"]  
   
 # Add to results dataframe  
 drop\_one\_results <- rbind(drop\_one\_results, data.frame(  
 dropped\_language = lang,  
 estimate = l2\_coef,  
 std\_error = l2\_se,  
 p\_value = l2\_p,  
 significant = l2\_p < 0.05  
 ))  
}  
  
# Sort results by p-value  
drop\_one\_results <- drop\_one\_results[order(drop\_one\_results$p\_value), ]  
  
# Look at summary statistics  
cat("Number of iterations where L2 effect is significant:", sum(drop\_one\_results$significant),   
 "out of", nrow(drop\_one\_results), "\n")

## Number of iterations where L2 effect is significant: 27 out of 31

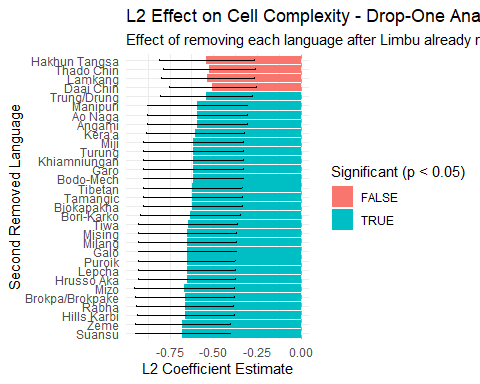
cat("Range of L2 coefficient estimates:",   
 round(min(drop\_one\_results$estimate), 3), "to",   
 round(max(drop\_one\_results$estimate), 3), "\n")

## Range of L2 coefficient estimates: -0.676 to -0.504

cat("Median p-value:", round(median(drop\_one\_results$p\_value), 3), "\n")

## Median p-value: 0.038

# Plot the results  
ggplot(drop\_one\_results, aes(x = reorder(dropped\_language, p\_value), y = estimate,   
 fill = significant)) +  
 geom\_bar(stat = "identity") +  
 geom\_errorbar(aes(ymin = estimate - std\_error, ymax = estimate + std\_error), width = 0.2) +  
 coord\_flip() +  
 labs(title = "L2 Effect on Cell Complexity - Drop-One Analysis",  
 subtitle = "Effect of removing each language after Limbu already removed",  
 x = "Second Removed Language",   
 y = "L2 Coefficient Estimate",  
 fill = "Significant (p < 0.05)") +  
 theme\_minimal() +  
 geom\_hline(yintercept = 0, linetype = "dashed", color = "darkgray")

 ### Complexity ~ Population

# 2. POPULATION SIZE EFFECTS  
cell\_pop\_mixed <- lmer(scaled\_cell\_complexity ~ scaled\_population + (1|subfamily), data = data\_sensitivity)  
form\_pop\_mixed <- lmer(scaled\_form\_complexity ~ scaled\_population + (1|subfamily), data = data\_sensitivity)  
case\_pop\_mixed <- lmer(scaled\_case\_marking\_complexity ~ scaled\_population + (1|subfamily), data = data\_sensitivity)  
  
# View population model summaries  
summary(cell\_pop\_mixed)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_cell\_complexity ~ scaled\_population + (1 | subfamily)  
## Data: data\_sensitivity  
##   
## REML criterion at convergence: 76.8  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -0.8671 -0.5545 -0.4723 0.0257 2.5385   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.05073 0.2252   
## Residual 0.61576 0.7847   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.1411 0.1694 6.2641 -0.833 0.435  
## scaled\_population -0.1571 0.1431 27.5589 -1.098 0.282  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## scald\_ppltn 0.042

summary(form\_pop\_mixed)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_form\_complexity ~ scaled\_population + (1 | subfamily)  
## Data: data\_sensitivity  
##   
## REML criterion at convergence: 73.1  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -0.9112 -0.4459 -0.3526 -0.1178 2.7689   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.06299 0.2510   
## Residual 0.53026 0.7282   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.18013 0.16650 8.85383 -1.082 0.308  
## scaled\_population -0.07211 0.13340 27.83491 -0.541 0.593  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## scald\_ppltn 0.052

summary(case\_pop\_mixed)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_case\_marking\_complexity ~ scaled\_population + (1 | subfamily)  
## Data: data\_sensitivity  
##   
## REML criterion at convergence: 90.8  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.3131 -0.5770 -0.4207 0.8920 2.2646   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.01203 0.1097   
## Residual 1.04756 1.0235   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.005017 0.189980 3.552319 -0.026 0.980  
## scaled\_population 0.097494 0.184445 27.520744 0.529 0.601  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## scald\_ppltn 0.014

### Complexity ~ agricultural intensity

# 3. AGRICULTURAL INTENSITY EFFECTS  
cell\_agr\_mixed <- lmer(scaled\_cell\_complexity ~ agriculture\_binary + (1|subfamily), data = data\_sensitivity)  
form\_agr\_mixed <- lmer(scaled\_form\_complexity ~ agriculture\_binary + (1|subfamily), data = data\_sensitivity)  
case\_agr\_mixed <- lmer(scaled\_case\_marking\_complexity ~ agriculture\_binary + (1|subfamily), data = data\_sensitivity)  
  
# View agriculture model summaries  
summary(cell\_agr\_mixed)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_cell\_complexity ~ agriculture\_binary + (1 | subfamily)  
## Data: data\_sensitivity  
##   
## REML criterion at convergence: 74.1  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -0.9266 -0.5349 -0.4118 0.1184 2.5367   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.07035 0.2652   
## Residual 0.58874 0.7673   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.05421 0.18545 8.08854 -0.292 0.777  
## agriculture\_binary -0.53310 0.38121 27.23589 -1.398 0.173  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## agrcltr\_bnr -0.325

summary(form\_agr\_mixed)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_form\_complexity ~ agriculture\_binary + (1 | subfamily)  
## Data: data\_sensitivity  
##   
## REML criterion at convergence: 69.4  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.0567 -0.4910 -0.3463 0.1822 2.7671   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.07557 0.2749   
## Residual 0.49102 0.7007   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.1039 0.1765 10.9369 -0.588 0.568  
## agriculture\_binary -0.5055 0.3495 27.7183 -1.446 0.159  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## agrcltr\_bnr -0.312

summary(case\_agr\_mixed)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_case\_marking\_complexity ~ agriculture\_binary + (1 | subfamily)  
## Data: data\_sensitivity  
##   
## REML criterion at convergence: 88.8  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.1234 -0.5670 -0.4232 0.8691 2.2656   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.01485 0.1219   
## Residual 1.04667 1.0231   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.04615 0.20748 5.38754 -0.222 0.832  
## agriculture\_binary 0.24719 0.50085 27.21492 0.494 0.626  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## agrcltr\_bnr -0.388

### complexity ~ political organization

# 4. POLITICAL ORGANIZATION EFFECTS  
cell\_pol\_mixed <- lmer(scaled\_cell\_complexity ~ political\_organization\_binary + (1|subfamily), data = data\_sensitivity)  
form\_pol\_mixed <- lmer(scaled\_form\_complexity ~ political\_organization\_binary + (1|subfamily), data = data\_sensitivity)  
case\_pol\_mixed <- lmer(scaled\_case\_marking\_complexity ~ political\_organization\_binary + (1|subfamily), data = data\_sensitivity)  
  
# View political organization model summaries  
summary(cell\_pol\_mixed)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_cell\_complexity ~ political\_organization\_binary + (1 |   
## subfamily)  
## Data: data\_sensitivity  
##   
## REML criterion at convergence: 74.5  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -0.90873 -0.46731 -0.38261 0.04705 2.52460   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.07574 0.2752   
## Residual 0.59906 0.7740   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.07532 0.18710 7.68914 -0.403 0.698  
## political\_organization\_binary -0.49228 0.42838 28.47455 -1.149 0.260  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## pltcl\_rgnz\_ -0.297

summary(form\_pol\_mixed)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_form\_complexity ~ political\_organization\_binary + (1 |   
## subfamily)  
## Data: data\_sensitivity  
##   
## REML criterion at convergence: 69.8  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.04073 -0.42603 -0.35824 0.08929 2.75118   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.07958 0.2821   
## Residual 0.49932 0.7066   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.1219 0.1778 10.4595 -0.685 0.508  
## political\_organization\_binary -0.4823 0.3934 28.5673 -1.226 0.230  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## pltcl\_rgnz\_ -0.288

summary(case\_pol\_mixed)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_case\_marking\_complexity ~ political\_organization\_binary +   
## (1 | subfamily)  
## Data: data\_sensitivity  
##   
## REML criterion at convergence: 88.9  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.1481 -0.5085 -0.4576 0.8419 2.2235   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.01201 0.1096   
## Residual 1.05747 1.0283   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.01270 0.20367 4.37581 -0.062 0.953  
## political\_organization\_binary 0.04881 0.55322 28.90087 0.088 0.930  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## pltcl\_rgnz\_ -0.350

## R-squared values for major analyses

### Complexity ~ Hill/Valley models

r.squaredGLMM(model\_case\_sensitivity)

## R2m R2c  
## [1,] 0.09347672 0.1235755

r.squaredGLMM(model\_cell\_sensitivity)

## R2m R2c  
## [1,] 0.1442393 0.2768161

r.squaredGLMM(model\_form\_sensitivity)

## R2m R2c  
## [1,] 0.1002318 0.2425052

r.squaredGLMM(model\_case)

## R2m R2c  
## [1,] 0.09278937 0.1219809

r.squaredGLMM(model\_cell)

## R2m R2c  
## [1,] 0.05093966 0.10617

r.squaredGLMM(model\_form)

## R2m R2c  
## [1,] 0.0216525 0.08268017

### Hill/Valley ~ ecology models

r.squaredGLMM(hill\_ecology\_model)

## R2m R2c  
## theoretical 0.02851554 0.02851554  
## delta 0.02297329 0.02297329

r.squaredGLMM(hill\_altitude)

## R2m R2c  
## theoretical 0.02729030 0.02729030  
## delta 0.02198079 0.02198079

r.squaredGLMM(hill\_slope)

## R2m R2c  
## theoretical 0.012391255 0.012391255  
## delta 0.009950803 0.009950803

### Complexity ~ sociocultural variables

#### L2\_status

r.squaredGLMM(cell\_L2\_mixed)

## R2m R2c  
## [1,] 0.1387773 0.2218777

r.squaredGLMM(form\_L2\_mixed)

## R2m R2c  
## [1,] 0.07924203 0.1890517

r.squaredGLMM(case\_L2\_mixed)

## R2m R2c  
## [1,] 0.0717251 0.1300981

#### population size

r.squaredGLMM(cell\_pop\_mixed)

## R2m R2c  
## [1,] 0.03677523 0.1100913

r.squaredGLMM(form\_pop\_mixed)

## R2m R2c  
## [1,] 0.008957825 0.114183

r.squaredGLMM(case\_pop\_mixed)

## R2m R2c  
## [1,] 0.009166445 0.02041192

#### agricultural\_intensity

r.squaredGLMM(cell\_agr\_mixed)

## R2m R2c  
## [1,] 0.05684786 0.15752

r.squaredGLMM(form\_agr\_mixed)

## R2m R2c  
## [1,] 0.05929465 0.1847704

r.squaredGLMM(case\_agr\_mixed)

## R2m R2c  
## [1,] 0.007981995 0.02186424

#### political\_organization

r.squaredGLMM(cell\_pol\_mixed)

## R2m R2c  
## [1,] 0.04003536 0.1477816

r.squaredGLMM(form\_pol\_mixed)

## R2m R2c  
## [1,] 0.04457714 0.175911

r.squaredGLMM(case\_pol\_mixed)

## R2m R2c  
## [1,] 0.0002586312 0.01148975

# Additional analyses

## Categorizing Split as Hill languages instead

# Alternative classification: Hill + Valley = 1, Split = 0  
data\_sensitivity$hill\_valley\_binary <- ifelse(data\_sensitivity$hill\_valley == "Split", 0, 1)  
  
# Fitting logistic regression models with alternative classification  
hill\_valley\_ecology\_model <- glm(  
 hill\_valley\_binary ~ scaled\_altitude + scaled\_stdev\_slope,  
 family = binomial,  
 data = data\_sensitivity  
)  
  
hill\_valley\_altitude <- glm(hill\_valley\_binary ~ scaled\_altitude, family = binomial, data = data\_sensitivity)  
hill\_valley\_slope <- glm(hill\_valley\_binary ~ scaled\_stdev\_slope, family = binomial, data = data\_sensitivity)  
  
# Summarizing model output  
summary(hill\_valley\_ecology\_model)

##   
## Call:  
## glm(formula = hill\_valley\_binary ~ scaled\_altitude + scaled\_stdev\_slope,   
## family = binomial, data = data\_sensitivity)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 1.1722 0.4524 2.591 0.00957 \*\*  
## scaled\_altitude 0.3266 0.6099 0.536 0.59229   
## scaled\_stdev\_slope 0.4950 0.4925 1.005 0.31492   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 35.403 on 30 degrees of freedom  
## Residual deviance: 32.753 on 28 degrees of freedom  
## AIC: 38.753  
##   
## Number of Fisher Scoring iterations: 4

summary(hill\_valley\_altitude)

##   
## Call:  
## glm(formula = hill\_valley\_binary ~ scaled\_altitude, family = binomial,   
## data = data\_sensitivity)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 1.1573 0.4498 2.573 0.0101 \*  
## scaled\_altitude 0.6167 0.5505 1.120 0.2626   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 35.403 on 30 degrees of freedom  
## Residual deviance: 33.847 on 29 degrees of freedom  
## AIC: 37.847  
##   
## Number of Fisher Scoring iterations: 4

summary(hill\_valley\_slope)

##   
## Call:  
## glm(formula = hill\_valley\_binary ~ scaled\_stdev\_slope, family = binomial,   
## data = data\_sensitivity)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 1.1409 0.4394 2.597 0.00942 \*\*  
## scaled\_stdev\_slope 0.6368 0.4356 1.462 0.14383   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 35.403 on 30 degrees of freedom  
## Residual deviance: 33.058 on 29 degrees of freedom  
## AIC: 37.058  
##   
## Number of Fisher Scoring iterations: 4

## Ecological variables as a predictor of complexity

## ALTITUDE MODELS  
# Altitude models - full dataset  
cell\_alt\_mixed <- lmer(scaled\_cell\_complexity ~ scaled\_altitude + (1|subfamily), data = data)  
form\_alt\_mixed <- lmer(scaled\_form\_complexity ~ scaled\_altitude + (1|subfamily), data = data)  
case\_alt\_mixed <- lmer(scaled\_case\_marking\_complexity ~ scaled\_altitude + (1|subfamily), data = data)  
  
# Altitude models - sensitivity dataset (without Limbu)  
cell\_alt\_mixed\_sens <- lmer(scaled\_cell\_complexity ~ scaled\_altitude + (1|subfamily), data = data\_sensitivity)  
form\_alt\_mixed\_sens <- lmer(scaled\_form\_complexity ~ scaled\_altitude + (1|subfamily), data = data\_sensitivity)  
case\_alt\_mixed\_sens <- lmer(scaled\_case\_marking\_complexity ~ scaled\_altitude + (1|subfamily), data = data\_sensitivity)  
  
# Slope variability models - full dataset  
cell\_slope\_mixed <- lmer(scaled\_cell\_complexity ~ scaled\_stdev\_slope + (1|subfamily), data = data)  
form\_slope\_mixed <- lmer(scaled\_form\_complexity ~ scaled\_stdev\_slope + (1|subfamily), data = data)  
case\_slope\_mixed <- lmer(scaled\_case\_marking\_complexity ~ scaled\_stdev\_slope + (1|subfamily), data = data)  
  
# Slope variability models - sensitivity dataset (without Limbu)  
cell\_slope\_mixed\_sens <- lmer(scaled\_cell\_complexity ~ scaled\_stdev\_slope + (1|subfamily), data = data\_sensitivity)  
form\_slope\_mixed\_sens <- lmer(scaled\_form\_complexity ~ scaled\_stdev\_slope + (1|subfamily), data = data\_sensitivity)  
case\_slope\_mixed\_sens <- lmer(scaled\_case\_marking\_complexity ~ scaled\_stdev\_slope + (1|subfamily), data = data\_sensitivity)  
  
# Get summaries for altitude models  
summary(cell\_alt\_mixed)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_cell\_complexity ~ scaled\_altitude + (1 | subfamily)  
## Data: data  
##   
## REML criterion at convergence: 92.3  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -0.90958 -0.57162 -0.32983 0.01346 2.95516   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.1647 0.4058   
## Residual 0.9084 0.9531   
## Number of obs: 32, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.007131 0.232016 3.962892 -0.031 0.977  
## scaled\_altitude 0.155745 0.200911 15.399117 0.775 0.450  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## scaled\_lttd -0.078

summary(form\_alt\_mixed)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_form\_complexity ~ scaled\_altitude + (1 | subfamily)  
## Data: data  
##   
## REML criterion at convergence: 92.3  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -0.8808 -0.4852 -0.3143 -0.0402 3.3478   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.1471 0.3835   
## Residual 0.9184 0.9583   
## Number of obs: 32, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.03005 0.22748 4.99586 -0.132 0.900  
## scaled\_altitude 0.10389 0.19946 17.05080 0.521 0.609  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## scaled\_lttd -0.074

summary(case\_alt\_mixed)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_case\_marking\_complexity ~ scaled\_altitude + (1 | subfamily)  
## Data: data  
##   
## REML criterion at convergence: 90.3  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.5343 -0.7179 -0.2910 0.7057 2.0940   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.0000 0.0000   
## Residual 0.9441 0.9716   
## Number of obs: 32, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) 3.406e-17 1.718e-01 3.000e+01 0.000 1.000  
## scaled\_altitude -2.939e-01 1.745e-01 3.000e+01 -1.684 0.103  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## scaled\_lttd 0.000   
## optimizer (nloptwrap) convergence code: 0 (OK)  
## boundary (singular) fit: see help('isSingular')

summary(cell\_alt\_mixed\_sens)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_cell\_complexity ~ scaled\_altitude + (1 | subfamily)  
## Data: data\_sensitivity  
##   
## REML criterion at convergence: 77.7  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -0.8599 -0.4725 -0.3464 -0.1464 2.5950   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.08761 0.2960   
## Residual 0.61837 0.7864   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.14413 0.18548 5.29516 -0.777 0.470  
## scaled\_altitude 0.06709 0.16287 16.90113 0.412 0.686  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## scaled\_lttd -0.047

summary(form\_alt\_mixed\_sens)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_form\_complexity ~ scaled\_altitude + (1 | subfamily)  
## Data: data\_sensitivity  
##   
## REML criterion at convergence: 73.2  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -0.8860 -0.3855 -0.3145 -0.1728 2.8065   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.07435 0.2727   
## Residual 0.52958 0.7277   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.18136 0.17136 8.10773 -1.058 0.320  
## scaled\_altitude 0.01211 0.15060 20.52459 0.080 0.937  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## scaled\_lttd -0.047

summary(case\_alt\_mixed\_sens)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_case\_marking\_complexity ~ scaled\_altitude + (1 | subfamily)  
## Data: data\_sensitivity  
##   
## REML criterion at convergence: 88.3  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.5090 -0.7135 -0.3019 0.7262 2.0703   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.0000 0.0000   
## Residual 0.9711 0.9854   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.01282 0.17703 29.00000 -0.072 0.943  
## scaled\_altitude -0.30273 0.17832 29.00000 -1.698 0.100  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## scaled\_lttd 0.022   
## optimizer (nloptwrap) convergence code: 0 (OK)  
## boundary (singular) fit: see help('isSingular')

# Get summaries for slope variability models  
summary(cell\_slope\_mixed)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_cell\_complexity ~ scaled\_stdev\_slope + (1 | subfamily)  
## Data: data  
##   
## REML criterion at convergence: 92.9  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -0.6795 -0.4898 -0.4171 -0.1373 3.1420   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.06212 0.2492   
## Residual 0.98284 0.9914   
## Number of obs: 32, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.001936 0.203240 3.698745 -0.010 0.993  
## scaled\_stdev\_slope -0.005075 0.183840 29.475291 -0.028 0.978  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## scld\_stdv\_s -0.046

summary(form\_slope\_mixed)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_form\_complexity ~ scaled\_stdev\_slope + (1 | subfamily)  
## Data: data  
##   
## REML criterion at convergence: 92.7  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -0.7419 -0.4581 -0.3716 -0.1416 3.4725   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.08806 0.2968   
## Residual 0.95805 0.9788   
## Number of obs: 32, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.01567 0.21103 5.03972 -0.074 0.944  
## scaled\_stdev\_slope -0.03473 0.18343 29.86068 -0.189 0.851  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## scld\_stdv\_s -0.059

summary(case\_slope\_mixed)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_case\_marking\_complexity ~ scaled\_stdev\_slope + (1 | subfamily)  
## Data: data  
##   
## REML criterion at convergence: 93  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.2287 -0.5028 -0.4748 0.8669 2.2600   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.002589 0.05089   
## Residual 1.030728 1.01525   
## Number of obs: 32, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) 7.529e-05 1.808e-01 3.054e+00 0.000 1.000  
## scaled\_stdev\_slope -1.954e-02 1.826e-01 2.807e+01 -0.107 0.916  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## scld\_stdv\_s -0.003

summary(cell\_slope\_mixed\_sens)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_cell\_complexity ~ scaled\_stdev\_slope + (1 | subfamily)  
## Data: data\_sensitivity  
##   
## REML criterion at convergence: 77.9  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -0.7348 -0.4422 -0.3806 -0.2434 2.5895   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.05627 0.2372   
## Residual 0.63914 0.7995   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.133888 0.174396 6.085167 -0.768 0.471  
## scaled\_stdev\_slope -0.006373 0.149622 28.864086 -0.043 0.966  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## scld\_stdv\_s -0.056

summary(form\_slope\_mixed\_sens)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_form\_complexity ~ scaled\_stdev\_slope + (1 | subfamily)  
## Data: data\_sensitivity  
##   
## REML criterion at convergence: 73.3  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -0.8653 -0.4364 -0.3698 -0.1579 2.8103   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.06335 0.2517   
## Residual 0.53457 0.7311   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.17261 0.16726 8.70458 -1.032 0.330  
## scaled\_stdev\_slope -0.03411 0.13816 28.98307 -0.247 0.807  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## scld\_stdv\_s -0.067

summary(case\_slope\_mixed\_sens)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_case\_marking\_complexity ~ scaled\_stdev\_slope + (1 | subfamily)  
## Data: data\_sensitivity  
##   
## REML criterion at convergence: 91  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.2014 -0.4951 -0.4607 0.8601 2.2303   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.004788 0.06919   
## Residual 1.063200 1.03112   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.00629 0.18767 2.80548 -0.034 0.976  
## scaled\_stdev\_slope -0.01864 0.18574 27.10243 -0.100 0.921  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## scld\_stdv\_s -0.002

## Languages miscategorized as Hill/Valley

# Check if there are Hill languages with below-average cell complexity  
hill\_languages <- filter(data, hill\_valley == "Hill") %>%  
 mutate(below\_average = scaled\_cell\_complexity < 0) %>%  
 arrange(scaled\_cell\_complexity)  
  
print("Hill languages sorted by cell complexity (lowest first):")

## [1] "Hill languages sorted by cell complexity (lowest first):"

print(select(hill\_languages, language, scaled\_cell\_complexity, below\_average))

## # A tibble: 18 × 3  
## language scaled\_cell\_complexity[,1] below\_average[,1]  
## <fct> <dbl> <lgl>   
## 1 Rabha -0.521 TRUE   
## 2 Suansu -0.521 TRUE   
## 3 Zeme -0.521 TRUE   
## 4 Galo -0.521 TRUE   
## 5 Milang -0.521 TRUE   
## 6 Mising -0.521 TRUE   
## 7 Hrusso Aka -0.521 TRUE   
## 8 Bjokapakha -0.521 TRUE   
## 9 Puroik -0.521 TRUE   
## 10 Miji -0.521 TRUE   
## 11 Brokpa/Brokpake -0.521 TRUE   
## 12 Tiwa -0.295 TRUE   
## 13 Kera'a 0.383 FALSE   
## 14 Hakhun Tangsa 1.12 FALSE   
## 15 Trung/Drung 1.51 FALSE   
## 16 Lamkang 1.63 FALSE   
## 17 Thado Chin 1.80 FALSE   
## 18 Daai Chin 2.14 FALSE

### Plotting

# Create classification based on theoretical expectations using mean (0)  
data\_with\_theoretical\_mismatches <- data %>%  
 mutate(  
 # Classify based on complexity relative to the mean (0)  
 theoretical\_status = case\_when(  
 # Hill languages with below-mean complexity  
 hill\_valley == "Hill" & scaled\_cell\_complexity < 0 ~   
 "Misclassified (Low Complexity Hill)",  
   
 # Valley languages with above-mean complexity  
 hill\_valley == "Valley" & scaled\_cell\_complexity > 0 ~   
 "Misclassified (High Complexity Valley)",  
   
 # Everything else matches theoretical expectations  
 TRUE ~ "Matches Theory"  
 )  
 )  
  
# Check the distribution of theoretical status  
print(table(data\_with\_theoretical\_mismatches$theoretical\_status))

##   
## Matches Theory Misclassified (High Complexity Valley)   
## 19 1   
## Misclassified (Low Complexity Hill)   
## 12

# Create scatter plot  
ggplot(data\_with\_theoretical\_mismatches, aes(x = scaled\_cell\_complexity, y = scaled\_case\_marking\_complexity)) +  
 # Adding quadrant lines at means (0 for scaled variables)  
 geom\_hline(yintercept = 0, linetype = "dashed", color = "gray") +  
 geom\_vline(xintercept = 0, linetype = "dashed", color = "gray") +  
   
 # Plot points  
 geom\_point(aes(  
 shape = hill\_valley,  
 color = theoretical\_status,  
 size = theoretical\_status  
 )) +  
   
 # Add labels for misclassified languages  
 geom\_text\_repel(  
 data = filter(data\_with\_theoretical\_mismatches, theoretical\_status != "Matches Theory"),  
 aes(label = language),  
 box.padding = 0.8,  
 point.padding = 0.5,  
 force = 8,  
 seed = 123  
 ) +  
   
 # Customize appearance  
 scale\_shape\_manual(values = c("Hill" = 17, "Valley" = 16, "Split" = 15)) +  
 scale\_color\_manual(values = c(  
 "Matches Theory" = "gray70",   
 "Misclassified (Low Complexity Hill)" = "#3366CC",  
 "Misclassified (High Complexity Valley)" = "#339900"  
 )) +  
 scale\_size\_manual(values = c(  
 "Matches Theory" = 2,   
 "Misclassified (Low Complexity Hill)" = 3.5,  
 "Misclassified (High Complexity Valley)" = 3.5  
 )) +  
   
 # Add labels  
 labs(  
 title = "Cell Complexity vs. Case Marking in Hill/Valley Languages",  
 x = "Cell Complexity (scaled)",  
 y = "Case Marking Complexity (scaled)",  
 shape = "Classification",  
 color = "Theory Alignment",  
 size = "Theory Alignment"  
 ) +  
   
 # Visual theme  
 theme\_minimal() +  
 theme(  
 panel.grid.minor = element\_blank(),  
 legend.position = "right"  
 )

