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 19  
## Split 7 0  
## Valley 6 0

#### 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: 89.5  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -0.78250 -0.58326 -0.48489 0.04758 3.11774   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.06428 0.2535   
## Residual 0.96939 0.9846   
## Number of obs: 32, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) 0.11362 0.24656 9.17566 0.461 0.656  
## hill\_valleySplit -0.50642 0.43820 24.06269 -1.156 0.259  
## hill\_valleyValley -0.07225 0.47102 28.80280 -0.153 0.879  
##   
## Correlation of Fixed Effects:  
## (Intr) hll\_vS  
## hll\_vllySpl -0.450   
## hll\_vllyVll -0.451 0.254

# 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: 90.2  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -0.7796 -0.4907 -0.4231 -0.1323 3.3631   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.06688 0.2586   
## Residual 0.99014 0.9951   
## Number of obs: 32, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) 0.02532 0.24953 11.28991 0.101 0.921  
## hill\_valleySplit -0.27948 0.44291 25.22968 -0.631 0.534  
## hill\_valleyValley 0.08086 0.47620 28.85620 0.170 0.866  
##   
## Correlation of Fixed Effects:  
## (Intr) hll\_vS  
## hll\_vllySpl -0.449   
## hll\_vllyVll -0.451 0.254

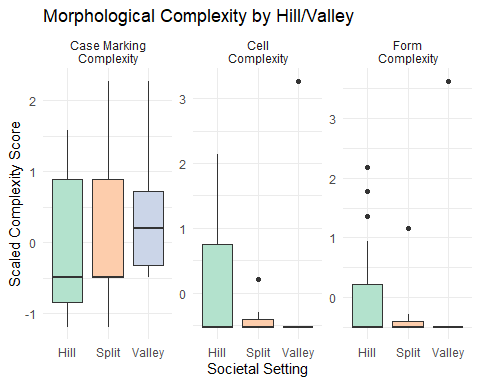
# Case marking complexity model  
model\_case <- lmer(scaled\_case\_marking\_complexity ~ hill\_valley + (1 | subfamily),  
 data = data)  
summary(model\_case)

## 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: 88.7  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.3581 -0.6965 -0.3002 0.6713 2.1040   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.05189 0.2278   
## Residual 0.94969 0.9745   
## Number of obs: 32, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.2099 0.2407 11.2404 -0.872 0.402  
## hill\_valleySplit 0.4118 0.4333 25.3164 0.950 0.351  
## hill\_valleyValley 0.6674 0.4646 28.8579 1.436 0.162  
##   
## Correlation of Fixed Effects:  
## (Intr) hll\_vS  
## hll\_vllySpl -0.460   
## hll\_vllyVll -0.457 0.254

# 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)

## 18   
## 18

# 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 × 49  
## 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 41 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>,  
## # agricultural\_intensity <fct>, agriculture\_comments <fct>,  
## # political\_organization <fct>, political\_organization\_comments <fct>, …

limbu\_form

## 23   
## 1.197211

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

## # A tibble: 1 × 49  
## 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 41 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>,  
## # agricultural\_intensity <fct>, agriculture\_comments <fct>,  
## # political\_organization <fct>, political\_organization\_comments <fct>, …

limbu\_cell

## 23   
## 1.024587

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

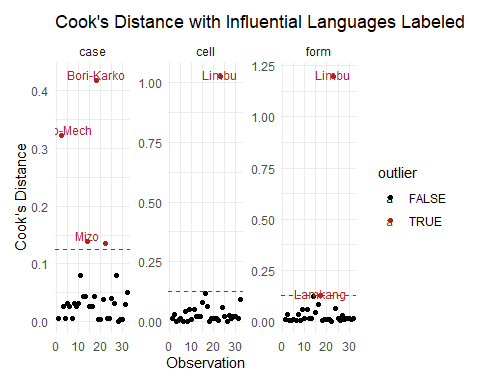
## # A tibble: 1 × 49  
## language pid glottocode mother\_node subfamily family reference alignment  
## <fct> <fct> <fct> <fct> <fct> <fct> <fct> <fct>   
## 1 Bori-Karko STET01 bori1243 Eastern Tani Macro-Ta… Sino-… Lalrempu… NOM-ACC   
## # ℹ 41 more variables: granular\_alignment <fct>, optional\_a <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>,  
## # agricultural\_intensity <fct>, agriculture\_comments <fct>,  
## # political\_organization <fct>, political\_organization\_comments <fct>, …

bori\_case

## 18   
## 0.4178971

## 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)



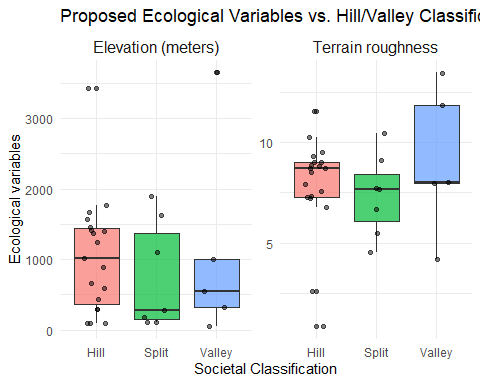
## Creating a new dataset without Limbu and Bodo-Mech

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: 71.9  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.2505 -0.6459 -0.3062 0.2462 2.3593   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.0943 0.3071   
## Residual 0.5419 0.7361   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)   
## (Intercept) 0.07013 0.20494 12.47712 0.342 0.7379   
## hill\_valleySplit -0.54069 0.32995 24.57779 -1.639 0.1140   
## hill\_valleyValley -0.69514 0.37975 26.23054 -1.831 0.0786 .  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Correlation of Fixed Effects:  
## (Intr) hll\_vS  
## hll\_vllySpl -0.380   
## hll\_vllyVll -0.368 0.248

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: 68.8  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.2428 -0.5462 -0.3749 0.2046 2.6120   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.08773 0.2962   
## Residual 0.48316 0.6951   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.03183 0.19474 15.13914 -0.163 0.872  
## hill\_valleySplit -0.32560 0.31167 25.54516 -1.045 0.306  
## hill\_valleyValley -0.60325 0.35888 26.75611 -1.681 0.104  
##   
## Correlation of Fixed Effects:  
## (Intr) hll\_vS  
## hll\_vllySpl -0.376   
## hll\_vllyVll -0.366 0.248

summary(model\_case\_sensitivity)

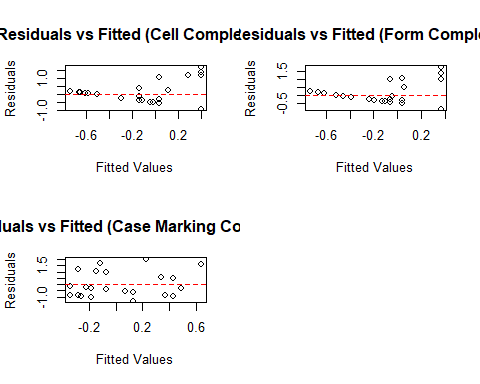
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [  
## lmerModLmerTest]  
## Formula: scaled\_case\_marking\_complexity ~ hill\_valley + (1 | subfamily)  
## Data: data\_sensitivity  
##   
## REML criterion at convergence: 86.6  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.3350 -0.7465 -0.3161 0.7643 2.0675   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.05368 0.2317   
## Residual 0.98106 0.9905   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.2080 0.2448 10.8295 -0.850 0.414  
## hill\_valleySplit 0.4127 0.4404 24.4924 0.937 0.358  
## hill\_valleyValley 0.7135 0.5029 26.1145 1.419 0.168  
##   
## Correlation of Fixed Effects:  
## (Intr) hll\_vS  
## hll\_vllySpl -0.459   
## hll\_vllyVll -0.419 0.242

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

## Testing assumptions

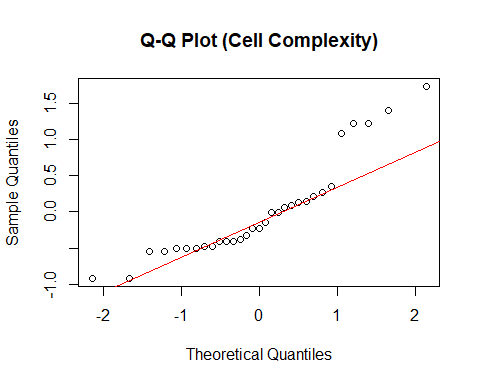
### Plotting residuals

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

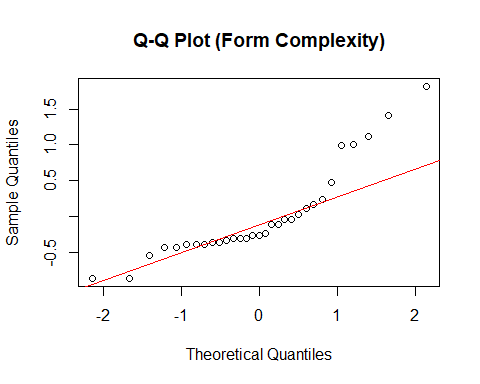
# Q-Q plot cell  
qqnorm(resid(model\_cell\_sensitivity), main = "Q-Q Plot (Cell Complexity)")  
qqline(resid(model\_cell\_sensitivity), col = "red")



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

##   
## Shapiro-Wilk normality test  
##   
## data: resid(model\_cell\_sensitivity)  
## W = 0.86717, p-value = 0.001203

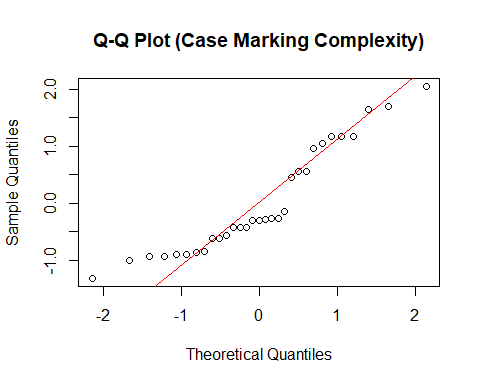
# Q-Q plot  
qqnorm(resid(model\_form\_sensitivity), main = "Q-Q Plot (Form Complexity)")  
qqline(resid(model\_form\_sensitivity), col = "red")



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

##   
## Shapiro-Wilk normality test  
##   
## data: resid(model\_form\_sensitivity)  
## W = 0.84152, p-value = 0.000336

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

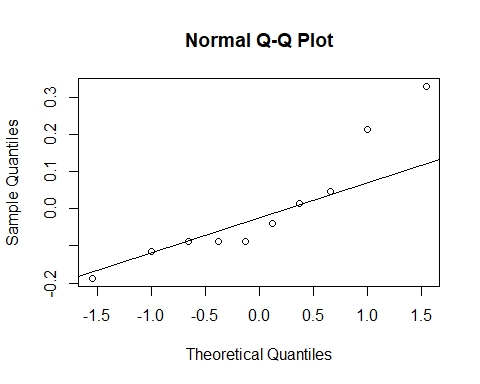


# Shapiro-Wilk test  
shapiro.test(resid(model\_case\_sensitivity))

##   
## Shapiro-Wilk normality test  
##   
## data: resid(model\_case\_sensitivity)  
## W = 0.8981, p-value = 0.006464

### 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 = 0.87464, p-value = 0.1132

## 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) 0.4714 0.3724 1.266 0.206  
## scaled\_altitude 0.2682 0.4446 0.603 0.546  
## scaled\_stdev\_slope -0.2161 0.4316 -0.501 0.617  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 41.381 on 30 degrees of freedom  
## Residual deviance: 40.952 on 28 degrees of freedom  
## AIC: 46.952  
##   
## 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) 0.4661 0.3705 1.258 0.208  
## scaled\_altitude 0.1585 0.3854 0.411 0.681  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 41.381 on 30 degrees of freedom  
## Residual deviance: 41.207 on 29 degrees of freedom  
## AIC: 45.207  
##   
## 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) 0.46017 0.36910 1.247 0.213  
## scaled\_stdev\_slope -0.08495 0.37231 -0.228 0.820  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 41.381 on 30 degrees of freedom  
## Residual deviance: 41.328 on 29 degrees of freedom  
## AIC: 45.328  
##   
## 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: 72.5  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.1184 -0.7293 -0.3337 0.1022 2.4658   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.0596 0.2441   
## Residual 0.5497 0.7414   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)   
## (Intercept) 0.07812 0.19616 12.71601 0.398 0.6971   
## L2 -0.58180 0.27921 27.93065 -2.084 0.0465 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## L2 -0.527

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: 69.7  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.12363 -0.58062 -0.32829 -0.00637 2.69696   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.06694 0.2587   
## Residual 0.49185 0.7013   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.03242 0.19066 15.54365 -0.170 0.867  
## L2 -0.40193 0.26520 28.18940 -1.516 0.141  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## L2 -0.510

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: 88.2  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.4385 -0.7310 -0.3425 0.8029 2.0070   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.07562 0.2750   
## Residual 0.96084 0.9802   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.1836 0.2511 8.3604 -0.731 0.485  
## L2 0.4707 0.3674 27.2310 1.281 0.211  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## L2 -0.547

#### 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: 90.9  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -0.7650 -0.5576 -0.3440 -0.1308 3.3656   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.0459 0.2143   
## Residual 0.9729 0.9863   
## Number of obs: 32, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) 0.1123 0.2430 9.1020 0.462 0.655  
## L2 -0.2860 0.3583 27.8497 -0.798 0.431  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## L2 -0.593

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: 91.3  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -0.7517 -0.4802 -0.3720 -0.1879 3.5310   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.08017 0.2831   
## Residual 0.96265 0.9811   
## Number of obs: 32, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) 0.01446 0.25255 10.98552 0.057 0.955  
## L2 -0.07966 0.35843 28.05613 -0.222 0.826  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## L2 -0.566

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: 90.2  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -1.4515 -0.7347 -0.3462 0.7613 2.0497   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.06321 0.2514   
## Residual 0.93579 0.9674   
## Number of obs: 32, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.1828 0.2445 9.1742 -0.748 0.473  
## L2 0.4561 0.3525 27.6585 1.294 0.206  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## L2 -0.577

#### 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: 14 out of 31

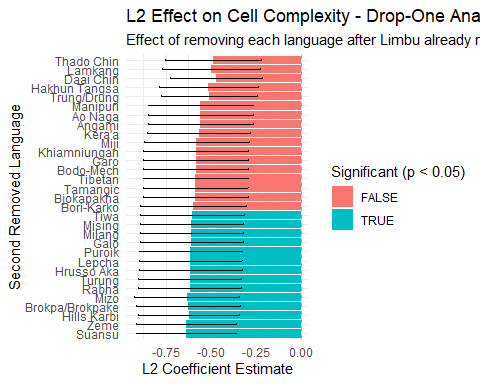
cat("Range of L2 coefficient estimates:", round(min(drop\_one\_results$estimate), 3),  
 "to", round(max(drop\_one\_results$estimate), 3), "\n")

## Range of L2 coefficient estimates: -0.638 to -0.469

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

## Median p-value: 0.055

# 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   
## -0.86549 -0.55792 -0.47075 0.02254 2.53898   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.0508 0.2254   
## Residual 0.6161 0.7849   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.1411 0.1695 6.2735 -0.832 0.436  
## scaled\_population -0.1558 0.1431 27.5294 -1.089 0.286  
##   
## 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   
## -0.9100 -0.4456 -0.3516 -0.1202 2.7696   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.0630 0.2510   
## Residual 0.5304 0.7283   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.18005 0.16652 8.85977 -1.081 0.308  
## scaled\_population -0.07117 0.13338 27.81357 -0.534 0.598  
##   
## 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   
## -1.3084 -0.5737 -0.4225 0.8937 2.2629   
##   
## Random effects:  
## Groups Name Variance Std.Dev.  
## subfamily (Intercept) 0.01215 0.1102   
## Residual 1.04798 1.0237   
## Number of obs: 31, groups: subfamily, 10  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)  
## (Intercept) -0.005045 0.190076 3.562308 -0.027 0.980  
## scaled\_population 0.095000 0.184483 27.478038 0.515 0.611  
##   
## 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,] 0.07157479 0.1197403

r.squaredGLMM(model\_cell\_sensitivity)

## R2m R2c  
## [1,] 0.1264166 0.2559057

r.squaredGLMM(model\_form\_sensitivity)

## R2m R2c  
## [1,] 0.08821983 0.2283333

### Hill/Valley ~ ecology models

r.squaredGLMM(hill\_ecology\_model)

## R2m R2c  
## theoretical 0.01837860 0.01837860  
## delta 0.01440316 0.01440316

r.squaredGLMM(hill\_altitude)

## R2m R2c  
## theoretical 0.007710643 0.007710643  
## delta 0.006028595 0.006028595

r.squaredGLMM(hill\_slope)

## R2m R2c  
## theoretical 0.002261112 0.002261112  
## delta 0.001765743 0.001765743

### Complexity ~ sociocultural variables

#### L2\_status

r.squaredGLMM(cell\_L2\_mixed)

## R2m R2c  
## [1,] 0.1198766 0.2059669

r.squaredGLMM(form\_L2\_mixed)

## R2m R2c  
## [1,] 0.06618585 0.1780519

r.squaredGLMM(case\_L2\_mixed)

## R2m R2c  
## [1,] 0.04979129 0.1191154

#### population size

r.squaredGLMM(cell\_pop\_mixed)

## R2m R2c  
## [1,] 0.03619123 0.1096094

r.squaredGLMM(form\_pop\_mixed)

## R2m R2c  
## [1,] 0.008725663 0.1139717

r.squaredGLMM(case\_pop\_mixed)

## R2m R2c  
## [1,] 0.008702663 0.02006298

#### 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) 1.2842 0.4498 2.855 0.0043 \*\*  
## scaled\_altitude 0.3127 0.5854 0.534 0.5932   
## scaled\_stdev\_slope 0.1567 0.4788 0.327 0.7434   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 33.118 on 30 degrees of freedom  
## Residual deviance: 32.333 on 28 degrees of freedom  
## AIC: 38.333  
##   
## 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) 1.2829 0.4496 2.853 0.00433 \*\*  
## scaled\_altitude 0.4006 0.5221 0.767 0.44287   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 33.118 on 30 degrees of freedom  
## Residual deviance: 32.441 on 29 degrees of freedom  
## AIC: 36.441  
##   
## 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) 1.2555 0.4378 2.867 0.00414 \*\*  
## scaled\_stdev\_slope 0.2919 0.4203 0.694 0.48742   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 33.118 on 30 degrees of freedom  
## Residual deviance: 32.638 on 29 degrees of freedom  
## AIC: 36.638  
##   
## 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: 19 × 3  
## language scaled\_cell\_complexity[,1] below\_average[,1]  
## <fct> <dbl> <lgl>   
## 1 Rabha -0.521 TRUE   
## 2 Turung -0.521 TRUE   
## 3 Suansu -0.521 TRUE   
## 4 Zeme -0.521 TRUE   
## 5 Galo -0.521 TRUE   
## 6 Milang -0.521 TRUE   
## 7 Mising -0.521 TRUE   
## 8 Hrusso Aka -0.521 TRUE   
## 9 Bjokapakha -0.521 TRUE   
## 10 Puroik -0.521 TRUE   
## 11 Miji -0.521 TRUE   
## 12 Brokpa/Brokpake -0.521 TRUE   
## 13 Tiwa -0.295 TRUE   
## 14 Kera'a 0.383 FALSE   
## 15 Hakhun Tangsa 1.12 FALSE   
## 16 Trung/Drung 1.51 FALSE   
## 17 Lamkang 1.63 FALSE   
## 18 Thado Chin 1.80 FALSE   
## 19 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)   
## 18 1   
## Misclassified (Low Complexity Hill)   
## 13

# 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"  
 )

