Reply to Koplenig

Data analysis

Henri Kauhanen, Sarah Einhaus & George Walkden

2022

1 Load data

```
data <- read.csv("../data/data.csv")
```

2 Descriptive statistics

2.1 General characteristics of the dataset

There are a total of

```
nrow(data)
```

```
## [1] 2143
```

languages in the dataset. However, not every language has data for each column of the data frame. The number of vehicular languages is

```
nrow(data[data$vehicularity==1, ])
```

```
## [1] 241
```

Of these,

```
nrow(data[data$vehicularity==1 & is.na(data$L2prop), ])
```

```
## [1] 152
```

do not have an L2 proportion estimate.

The number of non-vehicular languages is

```
nrow(data[data$vehicularity==0, ])
```

```
## [1] 1902
```

These all have an L2 proportion estimate, either real or imputed.

2.2 How many non-vehiculars have an imputed L2 proportion?

The number of non-vehicular languages with a zero L2 proportion is

```
nv0 <- nrow(data[data$vehicularity==0 & data$L2prop==0, ])
nv0
```

```
## [1] 1824
```

Of these, Ethnologue actually provides a numerical zero L2 proportion estimate for

```
nv0E <- nrow(data[data$vehicularity==0 & data$L2prop==0 & data$ethnologue_L2_users==TRUE, ])
nv0E
```

```
## [1] 4
```

languages. The rest have been imputed.

2.3 In how many cases is the data imputation wrong?

Ethnologue notes that the language is used as an L2 by speakers of some other set of languages (without giving numerical estimates) in

```
asL2 <- nrow(data[data$vehicularity==0 & data$L2prop==0 & !is.na(data$used_as_L2_by), ])
asL2
```

```
## [1] 404
```

of these cases. In other words, the data imputation is definitely wrong for

```
asL2/(nv0 - nv0E)
```

```
## [1] 0.221978
```

of the dataset.

3 Remove uncertain non-vehiculars

We now remove uncertain non-vehicular languages, i.e. all zero-L2-proportion non-vehicular languages except the

```
nv0E
```

```
## [1] 4
```

for which Ethnologue actually gives a zero L2 proportion estimate:

```
data2 <- rbind(data[data$vehicularity==1, ],
           data[data$vehicularity==0 & data$L2prop>0, ],
           data[data$vehicularity==0 & data$L2prop==0 & data$ethnologue_L2_users==TRUE, ])
```

There are

```
nrow(data2)
```

```
## [1] 323
```

languages in this subset of the original data. However, for some languages the L2 proportion estimate is not available. These are all vehicular languages (as indeed makes sense, for in Koplenig's data imputation scheme, uncertain non-vehicular languages always receive a zero L2 proportion estimate, not NA):

```
tmp <- data2[is.na(data2$L2prop), ]
nrow(tmp)
```

```
## [1] 152
```

```
table(tmp$vehicularity)
```

```
##
##   1
## 152
```

These languages therefore do not figure in the L2 proportion regressions, although they do figure in the population size regressions; for population size is available for all languages in the sample:

```
nrow(data2[is.na(data2$Population), ])
```

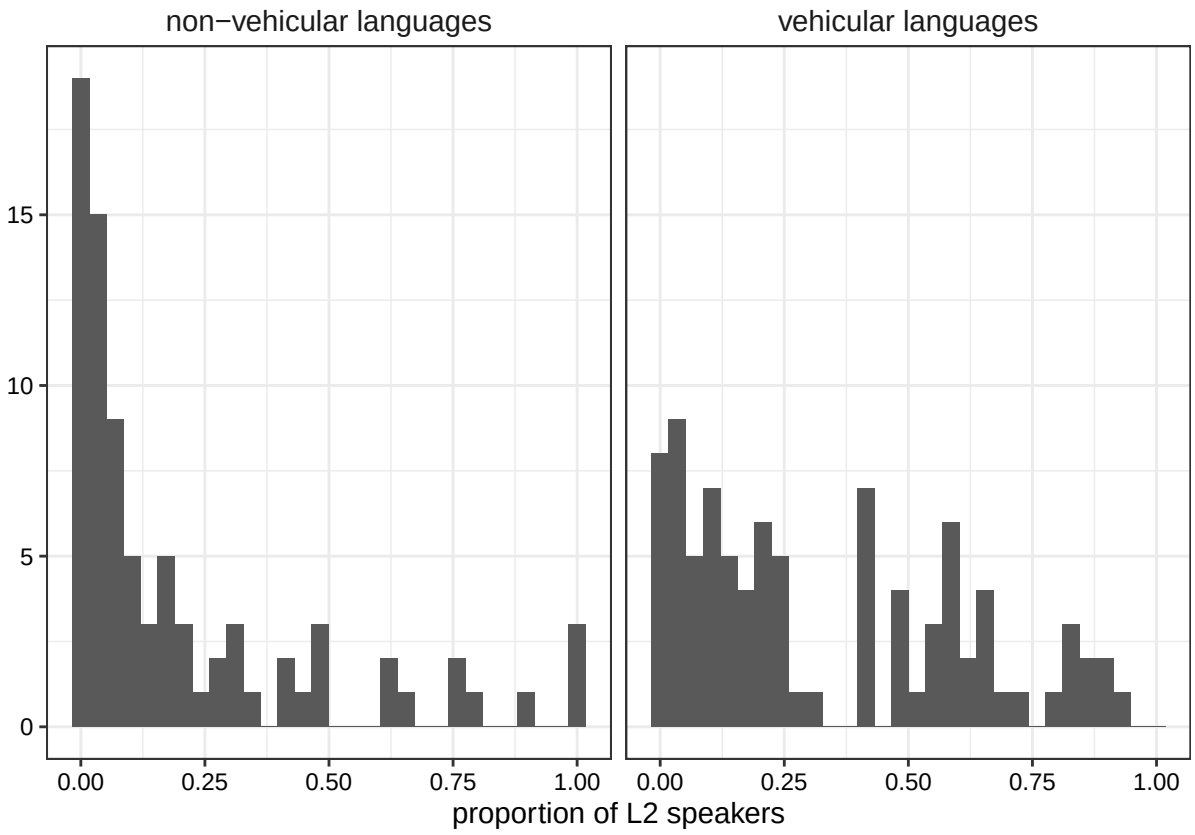
```
## [1] 0
```

4 Histogram of L2 speaker proportion

```
library(ggplot2)

# give nicer names to vehicularity column levels
datap <- data2
datap$vehicularity <- factor(datap$vehicularity,
                 labels=c("non-vehicular languages", "vehicular languages"))

# construct plot
g <- ggplot(datap, aes(x=L2prop)) + geom_histogram() + facet_wrap(.~vehicularity)
g <- g + theme_bw() + theme(axis.text=element_text(color="black"))
g <- g + theme(strip.background=element_blank(), strip.text=element_text(size=11))
g <- g + ylab("") + xlab("proportion of L2 speakers")

# save as pdf
pdf("../plots/histogram.pdf", height=3, width=6)
g
dev.off()
```

```
## pdf
##   2
```

```
# also print it here
g
```



5 Regressions

5.1 Morphological complexity

```
mod <- lm(MC~L2prop+log(Population), data2)
summary(mod)
```

```
##
## Call:
## lm(formula = MC ~ L2prop + log(Population), data = data2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.61212 -0.16519 -0.00423  0.18651  0.53496
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.845006   0.070888  11.920  < 2e-16 ***
## L2prop          -0.277717   0.079765  -3.482 0.000659 ***
## log(Population) -0.014384   0.004816  -2.987 0.003311 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2595 on 145 degrees of freedom
##   (175 observations deleted due to missingness)
## Multiple R-squared:  0.1082, Adjusted R-squared:  0.09587
```

```
## F-statistic: 8.794 on 2 and 145 DF,  p-value: 0.0002485
```

Adding an interaction does not improve model:

```
modb <- lm(MC~L2prop*log(Population), data2)
AIC(mod)
```

```
## [1] 25.66005
```

```
AIC(modb)
```

```
## [1] 25.9101
```

5.2 Morphological complexity, ≥ 6 features

```
mod6 <- lm(MC~L2prop+log(Population), data2[data2$NumChap>=6, ])
summary(mod6)
```

```
##
## Call:
## lm(formula = MC ~ L2prop + log(Population), data = data2[data2$NumChap >=
##     6, ])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.53436 -0.10991  0.02655  0.13920  0.48649
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.777517   0.068914  11.282  < 2e-16 ***
## L2prop          -0.250932   0.078095  -3.213  0.00178 **
## log(Population) -0.013336   0.004524  -2.948  0.00400 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2109 on 98 degrees of freedom
##   (132 observations deleted due to missingness)
## Multiple R-squared:  0.1432, Adjusted R-squared:  0.1257
## F-statistic: 8.192 on 2 and 98 DF,  p-value: 0.0005133
```

Again, adding an interaction does not improve model:

```
mod6b <- lm(MC~L2prop*log(Population), data2[data2$NumChap>=6, ])
AIC(mod6)
```

```
## [1] -22.79639
```

```
AIC(mod6b)
```

```
## [1] -20.84621
```

5.3 Information-theoretic complexity

```
modIC <- lm(H~L2prop+log(Population), data2)
summary(modIC)
```

```
##
## Call:
```

```
## lm(formula = H ~ L2prop + log(Population), data = data2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.49664 -0.13414 -0.03712  0.06457  0.74976
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      1.34410    0.16055   8.372 6.47e-13 ***
## L2prop          -0.24222    0.09890  -2.449  0.01624 *
## log(Population)  0.02703    0.01013   2.668  0.00904 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2566 on 91 degrees of freedom
##   (229 observations deleted due to missingness)
## Multiple R-squared:  0.1557, Adjusted R-squared:  0.1371
## F-statistic:  8.39 on 2 and 91 DF,  p-value: 0.0004527
```

Adding an interaction does not improve model:

```
modICb <- lm(H~L2prop*log(Population), data2)
AIC(modIC)
```

```
## [1] 15.95482
```

```
AIC(modICb)
```

```
## [1] 17.51845
```

6 Results plot

```
# construct first plot
g1 <- ggplot(data2, aes(x=L2prop, y=MC)) + geom_point() + geom_smooth(method=lm)
g1 <- g1 + xlab("proportion of L2 speakers") + ylab("morphological complexity")
g1 <- g1 + theme_bw()
g1 <- g1 + theme(axis.text=element_text(color="black"))
g1 <- g1 + ggtitle("A")

# construct second plot
g2 <- ggplot(data2, aes(x=L2prop, y=H)) + geom_point() + geom_smooth(method=lm)
g2 <- g2 + xlab("proportion of L2 speakers") + ylab("information-theoretic complexity")
g2 <- g2 + theme_bw()
g2 <- g2 + theme(axis.text=element_text(color="black"))
g2 <- g2 + ggtitle("B")

# construct third plot
g3 <- ggplot(data2, aes(x=log(Population), y=MC)) + geom_point() + geom_smooth(method=lm)
g3 <- g3 + xlab("log(population size)") + ylab("morphological complexity")
g3 <- g3 + theme_bw()
g3 <- g3 + theme(axis.text=element_text(color="black"))
g3 <- g3 + ggtitle("C")

# construct fourth plot
g4 <- ggplot(data2, aes(x=log(Population), y=H)) + geom_point() + geom_smooth(method=lm)
```

```
g4 <- g4 + xlab("log(population size)") + ylab("information-theoretic complexity")
g4 <- g4 + theme_bw()
g4 <- g4 + theme(axis.text=element_text(color="black"))
g4 <- g4 + ggtitle("D")

# pdf out
pdf("../plots/result.pdf", height=6, width=6)
grid.arrange(g1, g2, g3, g4, nrow=2, ncol=2)
dev.off()
```

```
## pdf
##   2
```

```
# print here, too
grid.arrange(g1, g2, g3, g4, nrow=2, ncol=2)
```