STAT 230A Final Project
Replication of Michalopoulos: The Origins of Ethnolinguistic Diversity

Andrej Leban, andrej_leban@berkeley.edu
Isaac Schmidt, ischmidt20@berkeley.edu

1 Paper Summary & Summary Statistics Table

1.1 Paper Summary

The paper by Michalopoulos (Michalopoulos 2012) aims to explain ethnolinguistic diversity within and across countries by assuming that a proxy quantity—the number of languages per square kilometer—is determined by a selection of various economic, historical, andgeographic variables. It determines that *variation in regional land quality* and *variation in elevation* are the most significant determinants of linguistic diversity. The hypothesis underpinning this examination is that differences in local land characteristics induce different levels of human capital across locations, which in turn, gives rise to localized ethnicities that are characterized by separate languages. The results of the empirical study presented are found to be consistent with this hypothesis.

The empirical results are obtained separately by three regressions:

- **Cross-country**: this takes the current political borders as the unit within which covariates such as the number of languages are counted.
- **Virtual countries**: To account for the arbitrary nature of some political boundaries with respect to ethnolinguistic groupings, the world is split into arbitrary *virtual countries* and the regression is performed again.
- **Adjacent regions**: To account for a potentially high "baseline" effect in some regions, adjacent regions are compared directly, which neutralizes region-specific fixed effects and focuses on the effect of the variables under consideration.

1.2 Exploratory Data Analysis and Summary Table

The data comes from multiple sources: the standard geographic data was sourced from the *Geographically Based Economic Data database*, the data on land quality for agriculture comes from *Ramankutty et al. (2002)*, and the data on the distribution of languages comes from the *World Language Mapping System*. Fortunately, the data provided by the author was already processed and cleaned to the extent used in the paper, so all we did was rename columns to more descriptive names.

The paper lacks a true summary table and shows a couple of EDA figures instead. We replicate two of those figures, and then display our own summary table of the features used in the paper's first regression—the *cross-country model*. Figure 1 shows the distribution of land suitability for agriculture across the world

at a resolution of .5-by.5 decimal degrees. The dependent variable represents the probability that a particular grid cell may be cultivated.

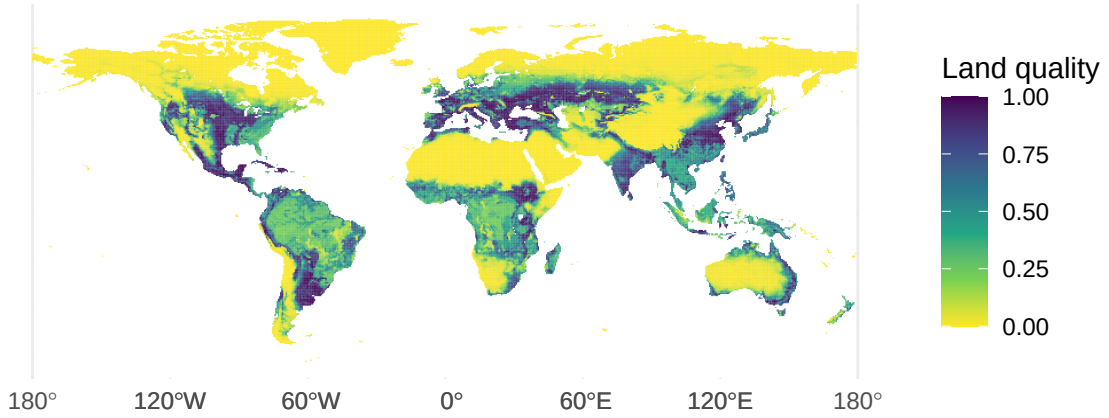

Figure 1: Land quality for agriculture across countries

Figure 2 shows the distribution of land quality within two countries selected in the paper—Greece and Nepal, obtained with a kernel density estimate using the Epanechnikov kernel.

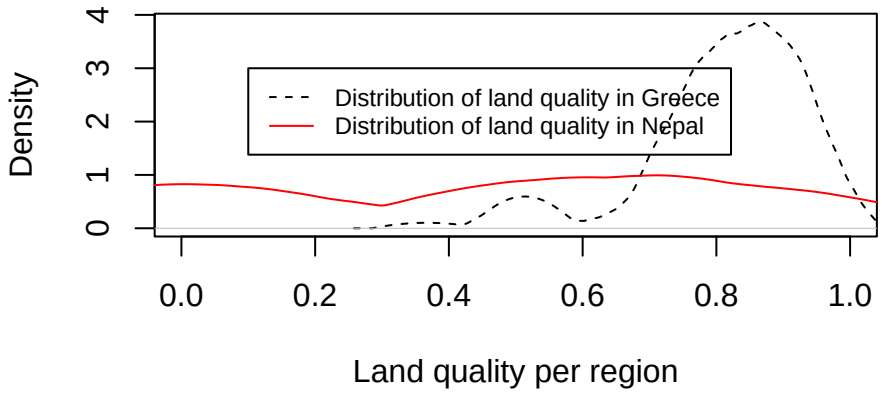

Figure 2: Kernel density of land quality in Greece and Nepal

Table 1 shows summary statistics of important variables for the first model. The dependent variable is `numLang`, which is the number of languages whose "traditional homeland" intersects with the country's boundary. Additional covariates are measures of centrality and variablity of the geographic data, the log of the country's 1995 population, human migration distance from Africa, and distance from a large body of water. While some other variables in the provided dataset have missing values for some countries, note that all variables included in the first regression are known for all countries.

2 Analysis 1: Cross-Country

The first model regresses the (log) number of languages within each country on the features described above. Michalopolous presents five different regression models, each containing a different number of covariates. The model, as described in the original paper, is the following:

$$\ln(\text{numLang}_i) = \beta_0 + \beta_1 * \text{absLat}_i + \beta_2 * \text{sdElev}_i + \beta_3 * \text{sdSuitable}_i + \beta_4 * X_i + \epsilon_i \quad (1)$$

Table 1: Summary statistics for covariates in cross-country analysis

	numLang	sdElev	sdSuitable	avgElev	avgSuitable	absLat	avgPrecip	avgTemp	lnArea	seaDist	migrationDist	lnPopDens1995
min	1.00	0.01	0.00	0.03	0.00	0.64	4.00	-6.37	-3.24	0.01	0.10	-10.22
median	10.00	0.25	0.18	0.42	0.44	24.18	77.11	20.93	0.61	0.18	5.79	-3.07
max	462.00	1.95	0.41	2.52	0.96	67.79	278.16	28.74	4.73	1.98	26.67	-0.25
mean	35.69	0.36	0.18	0.57	0.44	27.14	91.23	17.86	0.52	0.34	8.69	-3.27
sd	73.41	0.36	0.10	0.49	0.25	17.68	63.84	8.49	1.55	0.38	6.89	1.46

The first model only includes absolute latitude, the second model adds the mean and standard deviation of both elevation and land quality within each country, and the remaining models add additional covariates X_i.

2.1 Statement of Assumptions

The canonical assumptions of a linear model are that Equation 1 actually is the data-generation process, and that the error terms ϵ_i are normal with mean 0, and constant variance σ^2. Of course, these assumptions are rarely actually true, but fortunately, they can be relaxed slightly.

In the original paper, Michalopolous reported "robust" standard errors for the estimated coefficients, following Eicker-Huber-White's formula. The author used the default behavior of Stata's `robust` command, which includes the HC1 correction, as described in Section 6.4.1 of Peng Ding's lecture notes. Such standard errors relax the homoskedasticity requirement—$\text{Var}(\epsilon_i) = \sigma^2$—as well as the assumption of normality.

Thus, the only maintained assumptions are that the linear form in 1 holds, and that the error terms are independent with mean 0.

2.2 Replication

As the code and the data files were provided completely by the author, we were able to replicate the results perfectly. Table 2 perfectly replicates Table 1 in the original paper, and Table 3 displays additional information about each model. Note that all variables, including indicators, were standardized by Michalopolous, so we did so here as well. As mentioned above, the reported standard errors are follow the EHW formula, with HC1 correction, so they are generally slightly wider than what one would get from a homoskedastic model. Unsurprisingly, given the increasing number of features, the observed R^2 also increases with each model.

The interpretation of these results is much the same as in the original paper. In all four models, variation in elevation, and variation in land quality were useful predictors of the log number of languages, as originally hypothesized by the author.

The effects of the geographic variables are also noteworthy. Naturally, the effect of absolute latitude becomes insignificant once precipitation and temperature are introduced, as those two are highly correlated with distance from the equator. Between models 1.1 and 1.2, and also 1.2 and 1.3, the observed R^2 makes sizeable jumps, indicating that these geographic features are very useful in explaining linguistic diversity. About the distance distance from sea coefficient, the author has this intruiging interpretation:

Table 2: Main specification for the cross-country analysis. Italics indicate significance at the 1% level.

Variable	(1)	(2)	(3)	(4)	(5)
Variation in elevation		*0.310*	*0.256*	*0.291*	*0.275*
		(0.113)	*(0.079)*	*(0.089)*	*(0.101)*
Variation in land quality		*0.340*	*0.177*	*0.208*	*0.211*
		(0.084)	*(0.061)*	*(0.058)*	*(0.060)*
Mean elevation		-0.249	-0.111	-0.104	-0.085
		(0.113)	(0.106)	(0.118)	(0.113)
Mean land quality		-0.179	-0.069	-0.029	0.006
		(0.069)	(0.065)	(0.068)	(0.064)
Absolute latitude	*-0.479*	*-0.547*	-0.058	-0.033	-0.131
	(0.070)	*(0.061)*	(0.192)	(0.214)	(0.201)
Mean precipitation			*0.468*	*0.447*	*0.479*
			(0.086)	*(0.088)*	*(0.088)*
Mean temperature			0.270	0.385	0.404
			(0.197)	(0.213)	(0.183)
Ln(Area)			*0.517*	*0.482*	*0.464*
			(0.067)	*(0.073)*	*(0.074)*
Distance from the sea			0.053	0.063	0.073
			(0.065)	(0.062)	(0.064)
Migratory distance from Ethiopia			*-0.281*	-0.518	-0.513
			(0.063)	(0.199)	(0.218)
Ln(Population density in 1995)				-0.118	0.023
				(0.087)	(0.072)
Ln(Population density in 1500)					-0.235
					(0.105)
Year of independence					-0.108
					(0.066)
Timing of transition to agriculture					0.134
					(0.094)

Table 3: Information for each model in cross-country analysis.

Model	Continental Indicators	Observations	R^2
1	No	156	0.23
2	No	156	0.40
3	No	156	0.67
4	No	156	0.69
5	Yes	142	0.73

> ... areas that are increasingly isolated from the sea have been experiencing limited population mixing and thus should, on average, display higher ethnolinguistic fractionalization. It should be noted, however, that mean distance from the coast also captures the vulnerability of different areas to both the incidence and the intensity of invasion and colonization. Thus, the coefficient should be interpreted cautiously.

As the coefficient was never much more than one standard error above zero anyway, it is easy to ignore this effect entirely.

The final model introduces variables related to a country's history. The log of population density in 1500 does have a significant effect (at larger thresholds), and the author suggests "conditional on geographic characteristics, contemporary ethnic diversity may have been influenced by a country's historical levels of development." However, the other features entered here are insignificant, and even the sign of the density coefficient goes against intuition. It seems more natural that countries that were denser in 1500 would have greater ethnolinguistic diversity today, simply due to having more people to split, yet the model suggests the opposite. It could also be that this new feature is simply taking the effect of the 1990 density feature, as the two are very strongly correlated. Therefore, we consider it unlikely that such "historical levels of development" have much effect on modern diversity.

Another thing to note is that we noticed some inaccuracies with the provided years of independence. For example, the United States was listed as 1816, as opposed to 1776 or 1783. Other long-existing countries, such as Portugal and Denmark, were also given this 1816 value. Additionally, former Soviet republics were all (correctly) given a year of 1991, further emphasizing that independence year in general is almost arbitrary—it would have been surprising if it had a significant relationship with the outcome.

2.3 Critique of Assumptions

As stated before, this model only requires two main assumptions:

1. The functional form of 1 is correct.
2. The error terms are independent.

Both assumptions are hard to take fully for granted. Starting with the second, it is likely that there is some spatial correlation between neighboring countries, leading to dependence among the error terms. However, such correlation could have already been sufficiently modeled by including geographic features such as migration distance and average temperature. Additionally, the virtual country analysis, which we will reanalyze in Section 4, shows that the results of the model still hold after abstracting away from established country boundaries.

To informally test the whether or not the linearity assumption holds, Figure 3 shows the residual plot for the fifth model. While there is no curved relationship in the plot, it is clear that the residuals tend to increase as the dependent variable increases. This means that there is likely some other feature or combination of features which significantly impacts the log number of languages, which this model does not include. The residuals for the fifth model are what is plotted, but given it is the most specified model, it is expected that residual plots for the other models would look even worse.

Somewhat coincidentally, the residuals do look roughly homoskedastic, even though that assumption was relaxed. Another diagnostic plot that is commonly used is a Normal Q-Q plot of the residuals, but as this model does not require normality, we will not include such a plot here.

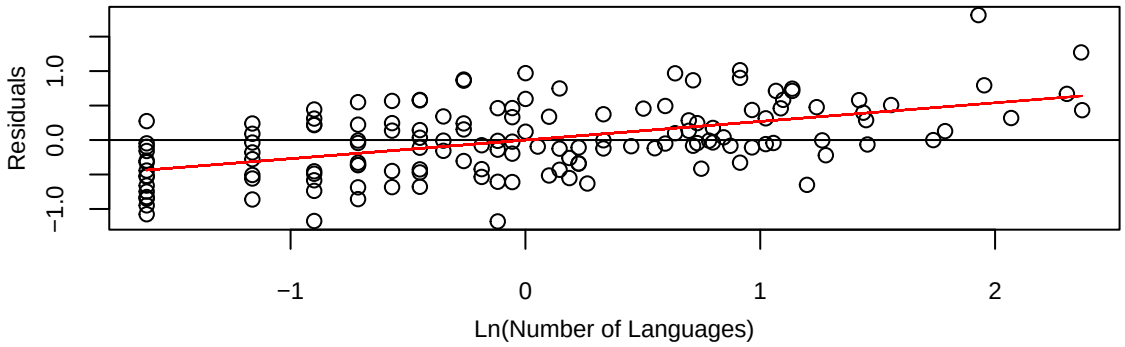

Figure 3: Residual Plot for Model 1.5

3 Robustness Check of Analysis 1

3.1 Table 2A

```
##
## Call:
## glm.nb(formula = numLang ~ absLat + sdSuitable + sdElev + avgElev +
##     avgSuitable + avgPrecip + avgTemp + lnArea + seaDist + migrationDist +
##     lnPopDens1995 + lnPopDens1500 + entryYear + agriTran + africa +
##     europe + americas + asiaPac, data = data, na.action = na.exclude,
##     init.theta = 1.934771646, link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2512  -1.0161  -0.4081   0.4434   3.5187
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)    6.548728   4.046319   1.618 0.105568
## absLat        -0.005966   0.017186  -0.347 0.728507
## sdSuitable     3.252227   0.967268   3.362 0.000773 ***
## sdElev         1.313302   0.413671   3.175 0.001500 **
## avgElev       -0.418495   0.328995  -1.272 0.203359
## avgSuitable    0.373707   0.401258   0.931 0.351679
## avgPrecip      0.010091   0.002181   4.626 3.73e-06 ***
## avgTemp        0.078466   0.030102   2.607 0.009143 **
## lnArea         0.528278   0.073553   7.182 6.86e-13 ***
## seaDist        0.253463   0.256217   0.989 0.322541
## migrationDist -0.157628   0.045867  -3.437 0.000589 ***
## lnPopDens1995  0.041208   0.093715   0.440 0.660141
## lnPopDens1500 -0.110614   0.095104  -1.163 0.244794
## entryYear     -0.003063   0.001842  -1.662 0.096436 .
## agriTran      -0.016956   0.063494  -0.267 0.789428
## africa        -0.308054   0.459205  -0.671 0.502322
## europe         0.130597   0.374891   0.348 0.727570
## americas       1.050666   0.678607   1.548 0.121558
## asiaPac        0.719299   0.382401   1.881 0.059971 .
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(1.9348) family taken to be 1)
##
##     Null deviance: 563.15  on 141  degrees of freedom
## Residual deviance: 144.04  on 123  degrees of freedom
##   (14 observations deleted due to missingness)
## AIC: 1112.7
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  1.935
##           Std. Err.:  0.250
##
##  2 x log-likelihood:  -1072.729
```

NOTE: Matches!

```
##
## Call:
## lm(formula = lnLang ~ absLat + dispElev + dispSuitable + avgElev +
##     avgSuitable + avgPrecip + avgTemp + lnArea + seaDist + migrationDist +
##     lnPopDens1995 + lnPopDens1500 + entryYear + agriTran + africa +
##     europe + americas + asiaPac, data = data, na.action = na.exclude)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.95035 -0.55786 -0.03237  0.52694  2.61099
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    5.852240   4.263804   1.373 0.172393
## absLat        -0.009741   0.019113  -0.510 0.611195
## dispElev       0.428598   0.134333   3.191 0.001802 **
## dispSuitable   1.314678   0.384770   3.417 0.000859 ***
## avgElev       -0.233907   0.326719  -0.716 0.475394
## avgSuitable   -0.046338   0.422776  -0.110 0.912901
## avgPrecip      0.010779   0.002425   4.445 1.94e-05 ***
## avgTemp        0.081696   0.033444   2.443 0.015995 *
## lnArea         0.351279   0.094939   3.700 0.000324 ***
## seaDist        0.269178   0.273772   0.983 0.327429
## migrationDist -0.126082   0.048136  -2.619 0.009920 **
## lnPopDens1995  0.059685   0.103261   0.578 0.564316
## lnPopDens1500 -0.226042   0.100970  -2.239 0.026973 *
## entryYear     -0.003279   0.001926  -1.702 0.091224 .
## agriTran       0.065991   0.067525   0.977 0.330345
## africa        -0.118792   0.497360  -0.239 0.811624
## europe         0.597281   0.403522   1.480 0.141385
## americas       0.375154   0.725772   0.517 0.606152
## asiaPac        0.457586   0.431350   1.061 0.290849
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8533 on 123 degrees of freedom
##   (14 observations deleted due to missingness)
## Multiple R-squared:  0.7323, Adjusted R-squared:  0.6931
## F-statistic: 18.69 on 18 and 123 DF,  p-value: < 2.2e-16
```

NOTE: matches or exceeds in significance, as well

```
##
## Call:
## lm(formula = lnLang ~ absLat + sdElev + sdClimate + avgElev +
##     climate + avgPrecip + avgTemp + lnArea + seaDist + migrationDist +
##     lnPopDens1995 + lnPopDens1500 + entryYear + agriTran + africa +
##     europe + americas + asiaPac, data = data, na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.9917 -0.5666 -0.0940  0.5144  2.7451
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    5.940308   4.284013   1.387  0.16806
## absLat        -0.011158   0.020088  -0.555  0.57959
## sdElev         1.045156   0.460917   2.268  0.02510 *
## sdClimate      2.505066   0.806127   3.108  0.00234 **
## avgElev       -0.230713   0.348637  -0.662  0.50936
## climate        0.660906   0.382811   1.726  0.08678 .
## avgPrecip      0.012137   0.002521   4.814 4.26e-06 ***
## avgTemp        0.078744   0.034420   2.288  0.02386 *
## lnArea         0.498980   0.078574   6.350 3.76e-09 ***
## seaDist        0.373275   0.274461   1.360  0.17631
## migrationDist -0.120027   0.048952  -2.452  0.01561 *
## lnPopDens1995  0.031198   0.101433   0.308  0.75893
## lnPopDens1500 -0.246419   0.102847  -2.396  0.01808 *
## entryYear     -0.003376   0.001917  -1.761  0.08077 .
## agriTran       0.095310   0.067456   1.413  0.16020
## africa        -0.162473   0.501039  -0.324  0.74628
## europe         0.624611   0.408883   1.528  0.12918
## americas       0.205573   0.728266   0.282  0.77821
## asiaPac        0.451329   0.428861   1.052  0.29468
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8583 on 123 degrees of freedom
##   (14 observations deleted due to missingness)
## Multiple R-squared:  0.7291, Adjusted R-squared:  0.6894
## F-statistic: 18.39 on 18 and 123 DF,  p-value: < 2.2e-16
```

NOTE: matches!

```
##
## Call:
## lm(formula = lnLang ~ absLat + sdElev + sdSoil + avgElev + soil +
##     avgPrecip + avgTemp + lnArea + seaDist + migrationDist +
##     lnPopDens1995 + lnPopDens1500 + entryYear + agriTran + africa +
##     europe + americas + asiaPac, data = data1.4, na.action = na.exclude)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.87797 -0.50870 -0.04869  0.49590  2.85469
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    4.324986   4.326770   1.000  0.31947
## absLat        -0.012498   0.019370  -0.645  0.52000
## sdElev         1.308730   0.439485   2.978  0.00350 **
## sdSoil         3.785250   1.349989   2.804  0.00587 **
## avgElev       -0.168396   0.353490  -0.476  0.63465
## soil           0.653355   0.499626   1.308  0.19342
## avgPrecip      0.012728   0.002396   5.313 4.88e-07 ***
## avgTemp        0.076052   0.033807   2.250  0.02625 *
## lnArea         0.502703   0.079264   6.342 3.92e-09 ***
## seaDist        0.250308   0.282993   0.885  0.37815
## migrationDist -0.112495   0.049314  -2.281  0.02426 *
## lnPopDens1995  0.061876   0.101543   0.609  0.54341
## lnPopDens1500 -0.244267   0.103564  -2.359  0.01992 *
## entryYear     -0.002706   0.001948  -1.389  0.16737
## agriTran       0.100822   0.068029   1.482  0.14089
## africa        -0.002027   0.497823  -0.004  0.99676
## europe         0.608691   0.403548   1.508  0.13403
## americas       0.117636   0.740684   0.159  0.87407
## asiaPac        0.412763   0.433711   0.952  0.34312
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.863 on 123 degrees of freedom
##   (14 observations deleted due to missingness)
## Multiple R-squared:  0.7261, Adjusted R-squared:  0.6861
## F-statistic: 18.12 on 18 and 123 DF,  p-value: < 2.2e-16
```

NOTE: matches!

3.2 Table 2B

```
TODO: write up a bit about the datasets used here
```

```
##
## Call:
## lm(formula = elf ~ abs_lat, data = data2bStd, na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -1.8470 -0.7628  0.1310  0.7781  1.9440
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.940e-17  7.800e-02   0.000        1
## abs_lat     -3.690e-01  7.827e-02  -4.714 5.76e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9327 on 141 degrees of freedom
## Multiple R-squared:  0.1362, Adjusted R-squared:   0.13
## F-statistic: 22.22 on 1 and 141 DF,  p-value: 5.764e-06
```

OK

```
##
## Call:
## lm(formula = elf ~ abs_lat + sd_emean + sd_climsuit + emean +
##     mean_climsuit, data = data2bStd, na.action = na.exclude)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.87650 -0.78488  0.01382  0.70933  2.08944
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.415e-16  7.605e-02   0.000  1.00000
## abs_lat       -4.337e-01  8.338e-02  -5.202 7.04e-07 ***
## sd_emean      -1.105e-01  1.199e-01  -0.922  0.35815
## sd_climsuit    2.938e-01  1.053e-01   2.789  0.00603 **
## emean          9.261e-02  1.113e-01   0.832  0.40670
## mean_climsuit  5.326e-02  8.675e-02   0.614  0.54024
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9094 on 137 degrees of freedom
## Multiple R-squared:  0.2021, Adjusted R-squared:  0.173
## F-statistic: 6.942 on 5 and 137 DF,  p-value: 8.292e-06
```

OK

```
##
## Call:
## lm(formula = elf ~ abs_lat + sd_emean + sd_climsuit + emean +
##     mean_climsuit + precav + tempav + lnareakm2 + distc + migdist +
##     lnpop95 + lpd1500 + yrentry + agritran + africa + europe +
##     americas + reg_eap, data = data2bStd, na.action = na.exclude)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.26948 -0.35908 -0.03879  0.46489  1.64637
##
```

```
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -9.380e-17  6.163e-02   0.000  1.00000
## abs_lat       -3.968e-01  2.931e-01  -1.354  0.17830
## sd_emean       3.630e-01  1.339e-01   2.712  0.00765 **
## sd_climsuit    2.309e-01  9.454e-02   2.442  0.01601 *
## emean         -3.009e-01  1.386e-01  -2.171  0.03187 *
## mean_climsuit  2.179e-01  1.093e-01   1.994  0.04834 *
## precav         1.798e-01  1.327e-01   1.355  0.17795
## tempav        -2.990e-02  2.454e-01  -0.122  0.90321
## lnareakm2      3.036e-02  1.129e-01   0.269  0.78852
## distc          2.815e-01  9.610e-02   2.929  0.00405 **
## migdist       -1.220e-01  2.781e-01  -0.439  0.66176
## lnpop95       -2.242e-02  1.208e-01  -0.186  0.85306
## lpd1500       -2.685e-01  1.343e-01  -1.999  0.04777 *
## yrentry        1.463e-01  9.602e-02   1.524  0.13002
## agritran      -7.968e-02  1.389e-01  -0.574  0.56726
## africa         8.347e-02  2.013e-01   0.415  0.67912
## europe         1.385e-01  1.490e-01   0.930  0.35441
## americas      -3.633e-01  2.268e-01  -1.602  0.11168
## reg_eap       -1.264e-01  1.157e-01  -1.093  0.27670
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.737 on 124 degrees of freedom
## Multiple R-squared:  0.5257, Adjusted R-squared:  0.4569
## F-statistic: 7.635 on 18 and 124 DF,  p-value: 5.106e-13
```

OK

```
##
## Call:
## lm(formula = elf3 ~ abs_lat + sd_emean + sd_climsuit + emean +
##     mean_climsuit + precav + tempav + lnareakm2 + distc + migdist +
##     lnpop95 + lpd1500 + yrentry + agritran + africa + europe +
##     americas + reg_eap, data = data2bStd, na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.5888 -0.5884 -0.1338  0.5310  2.0736
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -3.852e-16  7.373e-02   0.000  1.00000
## abs_lat       -1.156e-01  3.506e-01  -0.330  0.74220
## sd_emean       3.559e-01  1.601e-01   2.223  0.02804 *
## sd_climsuit    2.914e-01  1.131e-01   2.576  0.01116 *
## emean         -3.521e-01  1.658e-01  -2.123  0.03573 *
## mean_climsuit -1.414e-01  1.307e-01  -1.082  0.28146
## precav         4.553e-01  1.588e-01   2.868  0.00486 **
## tempav         2.476e-01  2.936e-01   0.843  0.40071
```

```
## lnareakm2     -2.473e-01  1.351e-01  -1.830  0.06963 .
## distc          4.520e-01  1.150e-01   3.931  0.00014 ***
## migdist       -5.348e-01  3.327e-01  -1.607  0.11051
## lnpop95       -9.337e-02  1.445e-01  -0.646  0.51945
## lpd1500       -2.393e-01  1.607e-01  -1.490  0.13886
## yrentry        5.755e-02  1.149e-01   0.501  0.61722
## agritran       1.545e-01  1.662e-01   0.930  0.35437
## africa        -1.706e-01  2.408e-01  -0.708  0.48006
## europe         1.397e-01  1.782e-01   0.784  0.43448
## americas       8.734e-02  2.713e-01   0.322  0.74805
## reg_eap        1.381e-01  1.384e-01   0.998  0.32021
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8816 on 124 degrees of freedom
## Multiple R-squared:  0.3212, Adjusted R-squared:  0.2227
## F-statistic:  3.26 on 18 and 124 DF,  p-value: 5.303e-05
```

OK

```
##
## Call:
## lm(formula = elf5 ~ abs_lat + sd_emean + sd_climsuit + emean +
##     mean_climsuit + precav + tempav + lnareakm2 + distc + migdist +
##     lnpop95 + lpd1500 + yrentry + agritran + africa + europe +
##     americas + reg_eap, data = data2bStd, na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.6182 -0.5485 -0.1193  0.6637  1.8739
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -2.995e-16  7.210e-02   0.000 1.000000
## abs_lat       -1.851e-01  3.429e-01  -0.540 0.590204
## sd_emean       4.723e-01  1.566e-01   3.016 0.003104 **
## sd_climsuit    2.928e-01  1.106e-01   2.647 0.009165 **
## emean         -4.749e-01  1.622e-01  -2.928 0.004055 **
## mean_climsuit -6.188e-02  1.279e-01  -0.484 0.629229
## precav         4.043e-01  1.553e-01   2.604 0.010330 *
## tempav         1.807e-01  2.871e-01   0.629 0.530190
## lnareakm2     -1.862e-01  1.321e-01  -1.409 0.161262
## distc          4.138e-01  1.124e-01   3.681 0.000346 ***
## migdist       -2.804e-01  3.254e-01  -0.862 0.390511
## lnpop95        5.586e-03  1.413e-01   0.040 0.968537
## lpd1500       -2.137e-01  1.571e-01  -1.360 0.176243
## yrentry        1.107e-01  1.123e-01   0.985 0.326432
## agritran       1.956e-01  1.625e-01   1.203 0.231118
## africa        -8.903e-04  2.355e-01  -0.004 0.996990
## europe         1.646e-01  1.743e-01   0.945 0.346619
## americas      -1.426e-01  2.653e-01  -0.537 0.591952
```

```
## reg_eap        3.300e-02  1.353e-01   0.244 0.807742
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8622 on 124 degrees of freedom
## Multiple R-squared:  0.3509, Adjusted R-squared:  0.2567
## F-statistic: 3.724 on 18 and 124 DF,  p-value: 6.471e-06
```

OK

```
##
## Call:
## lm(formula = elf7 ~ abs_lat + sd_emean + sd_climsuit + emean +
##     mean_climsuit + precav + tempav + lnareakm2 + distc + migdist +
##     lnpop95 + lpd1500 + yrentry + agritran + africa + europe +
##     americas + reg_eap, data = data2bStd, na.action = na.exclude)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.60444 -0.50238  0.02369  0.57650  1.80411
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -4.751e-16  6.706e-02   0.000 1.000000
## abs_lat       -6.412e-02  3.189e-01  -0.201 0.840982
## sd_emean       4.260e-01  1.456e-01   2.925 0.004093 **
## sd_climsuit    2.154e-01  1.029e-01   2.094 0.038278 *
## emean         -4.620e-01  1.508e-01  -3.063 0.002688 **
## mean_climsuit -2.134e-01  1.189e-01  -1.795 0.075131 .
## precav         4.871e-01  1.444e-01   3.373 0.000991 ***
## tempav         3.158e-01  2.670e-01   1.183 0.239188
## lnareakm2     -1.745e-01  1.229e-01  -1.420 0.158115
## distc          3.264e-01  1.046e-01   3.121 0.002240 **
## migdist       -3.594e-01  3.026e-01  -1.188 0.237242
## lnpop95        7.032e-03  1.314e-01   0.053 0.957421
## lpd1500       -2.111e-01  1.461e-01  -1.445 0.151068
## yrentry        6.106e-02  1.045e-01   0.585 0.559945
## agritran       3.381e-01  1.511e-01   2.237 0.027073 *
## africa         9.670e-03  2.190e-01   0.044 0.964855
## europe         1.604e-02  1.621e-01   0.099 0.921321
## americas      -1.658e-01  2.468e-01  -0.672 0.502771
## reg_eap       -2.955e-03  1.259e-01  -0.023 0.981309
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8019 on 124 degrees of freedom
## Multiple R-squared:  0.4385, Adjusted R-squared:  0.357
## F-statistic:  5.38 on 18 and 124 DF,  p-value: 4.415e-09
```

OK

```
##
```

```
## Call:
## lm(formula = elf9 ~ abs_lat + sd_emean + sd_climsuit + emean +
##     mean_climsuit + precav + tempav + lnareakm2 + distc + migdist +
##     lnpop95 + lpd1500 + yrentry + agritran + africa + europe +
##     americas + reg_eap, data = data2bStd, na.action = na.exclude)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.66758 -0.54364  0.05585  0.54828  1.74016
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -2.912e-16  6.471e-02   0.000  1.00000
## abs_lat       -1.244e-01  3.078e-01  -0.404  0.68667
## sd_emean       4.133e-01  1.406e-01   2.941  0.00391 **
## sd_climsuit    1.561e-01  9.928e-02   1.572  0.11839
## emean         -3.674e-01  1.456e-01  -2.524  0.01286 *
## mean_climsuit -2.386e-02  1.148e-01  -0.208  0.83562
## precav         3.755e-01  1.394e-01   2.694  0.00804 **
## tempav         3.019e-01  2.577e-01   1.172  0.24354
## lnareakm2     -1.462e-02  1.186e-01  -0.123  0.90207
## distc          2.290e-01  1.009e-01   2.269  0.02497 *
## migdist       -5.396e-01  2.921e-01  -1.847  0.06706 .
## lnpop95       -9.025e-02  1.269e-01  -0.711  0.47817
## lpd1500       -1.662e-01  1.410e-01  -1.179  0.24084
## yrentry        4.645e-02  1.008e-01   0.461  0.64579
## agritran       1.554e-01  1.459e-01   1.065  0.28881
## africa        -1.430e-01  2.114e-01  -0.677  0.49991
## europe        -4.793e-02  1.564e-01  -0.306  0.75980
## americas      -2.222e-01  2.381e-01  -0.933  0.35264
## reg_eap       -8.756e-02  1.215e-01  -0.721  0.47235
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7739 on 124 degrees of freedom
## Multiple R-squared:  0.477,  Adjusted R-squared:  0.4011
## F-statistic: 6.284 on 18 and 124 DF,  p-value: 1.027e-10
```

OK

4 Analysis 2: Virtual Country (Re-Analysis)

The second analysis Michalopolous presents in his paper is essentially a repeat of the the first, but instead aggregating geographic and ethnic information over "virtual" countries instead of real ones. The stated motivation for this is "to investigate whether the relationship between geography and ethnic diversity holds true at an arbitrary level of aggregation."

As with the previous analysis, the geographic features are derived from a dataset of cells, each of size .5-by.5 decimal degrees. However, instead of aggregating these cells at the country level as before, we now split up the world into blocks of size 2.5-by-2.5 decimal degrees, with each block containing 25 cells. Each

block is precisely a "virtual country."

To obtain the number of languages in each virtual country, Michalopolous simply intersected the shapefile provided in the World Language Mapping System with the newly-formed grid. However, probably due to the proprietary nature of the WLMS, the "number of languages" variable was withheld from the public data download, meaning we could not exactly replicate the analysis.

Fortunately, we stumbled across the *Geo-referencing of Ethnic Groups* (GREG) dataset (Weidmann, Rød, and Cederman 2010), which contains a shapefile of the locations of 928 ethnic groups across the world. As Michalopolous was only using linguistic diversity as a proxy for ethnic diversity, we decided it would be useful to model ethnic diversity directly, to see if the original paper's results held up with the GREG data.

4.1 Data Cleaning

The original GREG dataset required some manipulation to get it in a format suitable to swap in for the WLMS. Each polygonwas labeled with up to three ethnic groups, so we had to melt and then dissolve the polygons such that each polygon represented only one ethnic group, and each ethnic group was only assigned to one polygon. For details, see the appendix, which shows the geoprocessing steps performed with the `geopandas` module in Python.

In the original paper, Michalopolous described the steps he took to filter the virtual countries, on criteria mostly based on the amount of "coverage" each country had in the WLMS data. If a large portion of a virtual country was an area which contained no languages—for example, the Sahara Desert—that virtual country was excluded from the analysis. The public data download, which contained the virtual countries after this filter had been applied, contained 1,888 virtual countries. Due to differences in coverage between WLMS and GREG, applying the same criteria to GREG would have resulted in 2,476 countries. Including these additional countries would have required obtaining the other features for these areas, and as Michalopolous did not document this procedure well, we decided that this was not feasible. The end result was that our dataset only included the intersection of the sets derived WLMS and GREG, which excluded the ~600 countries from the dataset derived from GREG, but also about 30 countries that had enough coverage in WLMS, but did not in GREG.

Finally, the actual regressions performed in the paper used a dataset that was further filtered down. That is, there must have been at least 3000 people living in the virtual country in 1995, and at least 10 of the 25 cells that comprise a virtual country had to have been completely covered by the WLMS dataset. We applied both of these criteria here as well when reproducing the regressions.

4.2 Replication

The model specification for this analysis is almost exactly the same as before, except now each unit i is a virtual country, and "numLang" is really the number of unique ethnic groups:

$$\ln(\text{numLang}_i) = \beta_0 + \beta_1 * \text{absLat}_i + \beta_2 * \text{sdElev}_i + \beta_3 * \text{sdSuitable}_i + \beta_4 * X_i + \epsilon_i \tag{2}$$

Additionally, regressions 2.5, 2.6, and 2.7 are performed only on virtual countries meeting a certain criterion. Regression 2.5 looks only at virtual countries located in the tropics, 2.6 looks at countries not located in the tropics, and 2.7 filters to virtual countries that are located entirely within a real country.

Table 4: Main specification for the virtual country analysis. Italics indicate significance at the 1% level.

Variable	(1)	(2)	(3)	(4)	(5)	(6)	(7)
Variation in elevation		*0.156*	*0.130*	*0.089*	-0.058	*0.158*	*0.145*
		(0.056)	*(0.047)*	*(0.033)*	(0.050)	*(0.042)*	*(0.041)*
Variation in land quality		*0.202*	*0.136*	*0.180*	0.103	*0.213*	*0.180*
		(0.054)	*(0.045)*	*(0.030)*	(0.053)	*(0.038)*	*(0.042)*
Mean elevation		-0.059	-0.111	*-0.138*	0.065	-0.108	-0.176
		(0.042)	(0.064)	*(0.051)*	(0.161)	(0.079)	(0.079)
Mean land quality		0.042	0.065	0.072	0.068	0.093	0.071
		(0.076)	(0.039)	(0.037)	(0.061)	(0.046)	(0.056)
Absolute latitude	*-0.196*	-0.146	*-0.382*	*-0.553*	-0.070	-0.154	*-0.728*
	(0.063)	(0.089)	*(0.144)*	*(0.162)*	(0.109)	(0.190)	*(0.265)*
Mean precipitation			*0.162*	0.036	0.134	-0.033	-0.026
			(0.056)	(0.046)	(0.108)	(0.048)	(0.070)
Mean temperature			-0.332	*-0.472*	0.111	-0.229	-0.492
			(0.150)	*(0.136)*	(0.174)	(0.178)	(0.215)
Ln(Area)			-0.096	0.001	0.013	0.030	-0.005
			(0.056)	(0.029)	(0.058)	(0.033)	(0.041)
Distance from the sea			*0.208*	*0.156*	*0.179*	*0.149*	*0.206*
			(0.044)	*(0.029)*	*(0.065)*	*(0.035)*	*(0.038)*
Water area			-0.006	-0.027	-0.066	-0.007	-0.013
			(0.028)	(0.023)	(0.035)	(0.031)	(0.038)
Within-country indicator			-0.091	-0.088	0.025	*-0.151*	
			(0.054)	(0.040)	(0.063)	*(0.050)*	
Number of countries			*0.293*	*0.260*	*0.294*	*0.229*	
			(0.038)	*(0.044)*	*(0.066)*	*(0.059)*	
Migratory distance from Ethiopia			-0.118	-0.378	-0.991	-0.118	-0.482
			(0.091)	(0.175)	(0.716)	(0.223)	(0.252)
Ln(Population density in 1995)				0.008	*0.267*	-0.051	0.009
				(0.039)	*(0.070)*	(0.049)	(0.053)

Table 5: Information for each model in virtual country analysis.

Model	Country Indicators	Observations	WLMS R^2	GREG R^2
1	No	1449	0.31	0.04
2	No	1449	0.36	0.12
3	No	1449	0.53	0.34
4	Yes	1449	0.70	0.56
5	Yes	447	0.73	0.65
6	Yes	1002	0.56	0.54
7	Yes	860	0.66	0.49

Table 4 shows the results of regressing the log of number of ethnic groups on different sets of features, reproducing Table 4 of the original paper. Note that the features here are very similar to the ones used in the cross-country analysis. The only differences are that features directly relating to real countries, such as independence year, have been swapped for features describing the position of the virtual country in relation to real countries. Table 5 shows information about each model, including the observed R^2 in the original WLMS regression, and our GREG regression.

Here, the reported standard errors are cluster-robust, with the clusters defined by the real country in which the centroid of each virtual country falls. Whether Stata applies any corrections by default is unclear, but the base formula should be the same as the one described in Section 24.4.1 of Peng Ding's lecture notes, which we implemented in R using the `sandwich` and `lmtest` packages. Michalopolous did not justify his decision to cluster by real country, and the reader would not know that he did so without checking the footnotes or his code. However, it seems a reasonable decision, considering that virtual countries within the same real country are certainly related beyond any similarities in their features.

Additionally, models 2.4 through 2.7 use country fixed effects. Again, exactly how to replicate the Stata code was not obvious, but the `lm_robust` function from the `estimatr` package appeared to work. This technique essentially just includes one indicator variable for each real country in the model, and then ignores the estimates for those variables in the output. Per Michalopolous:

> Such inclusion of powerful controls, not possible in a cross-country framework, allows me to explicitly take into account any systematic elements related to the nation-building process of current states and thus produce reliable estimates of the effect of geographic heterogeneity on ethnic diversity.

4.3 Comparison

One major difference between GREG and WLMS is that the footprint of each unique ethnic group in GREG are larger than that for each unique language in the WLMS. This makes sense, considering that GREG only contains about 900 entries, yet there are a few thousand unique languages in the WLMS. As a result, the dependent variable is generally a lot smaller in our replication compared to that in the original paper. Michalopolous reports the median number of languages per virtual country as 3, yet here, more than half of the virtual countries contain only one ethnic group. A possible solution to adjust for this would be to simply reduce the size of the virtual countries, but this was infeasible due to our inability to recalculate the rest of the features.

As for the actual results, the first noticeable difference is that of the R^2 coefficients. In all models, the R^2 is considerably lower in our replication compared to the original, although the differences are smaller for the models including country fixed effects. For example, absolute latitude on its own explains just 4% of the variation in log number of ethnic groups, compared to 33% in the original. However, the coefficient is still significant at the 1% level and ha negative sign, as it does in the original.

Beyond the worse fits overall, the inference around the coefficients does not differ much between the two models. If a variable is significant at the 1% level in one model, there is a good chance it is significant at at least the 5% or 10% level in its counterpart, or vice versa. Variation in elevation and variation in land quality continue to have a large relationship with the outcome, leading further evidence to the author's original hypothesis. The number of real countries intersected by a virtual country is also a really strong predictor in both sets of models. Michalopolous ponders that this "may be suggestive of the effect of state formation on ethnic diversity and/or an artifact of modern states having drawn political borders

along ethnic boundaries." This certainly seems reasonable, but given how diminished the effects of some of the other variables are in our replication, it is surprising that this one is still so large. Perhaps this is another artifact of the aforementioned reduced granularity of GREG compared to WLMS, that ethnic groups are even more strongly correlated with national boundaries than languages are.

However, there are a few noteworthy inconsistencies. One is that distance from the sea is significant for every model here, but only for model 2.5 (tropical locations) in the original. Another oddity of the tropical model is that the sign for variation in elevation flips to negative, and log of population density becomes a strong predictor. Without speculating on real-world phenomena, one reason for this could be differences in how virtual countries in tropical areas were filtered in the GREG dataset compared to WLMS.

4.4 Robustness Check

5 References

Ding, Peng. 2022. "Linear Model and Extensions." University of California, Berkeley.

Michalopoulos, Stelios. 2012. "The Origins of Ethnolinguistic Diversity." *American Economic Review* 102 (4): 1508–39. https://doi.org/10.1257/aer.102.4.1508.

Weidmann, Nils B., Jan Ketil Rød, and Lars-Erik Cederman. 2010. "Representing Ethnic Groups in Space: A New Dataset." *Journal of Peace Research* 47 (4): 491–99. https://doi.org/10.1177/0022343310368352.

6 Appendix: code

```
cells = read_sf(dsn = 'data_raw/Virtual_country', layer = 'virtual_cntrygrid')
countries = read_sf(dsn = 'countries', layer = 'countries')
data = read.dta13("data_raw/Tables1-3a.dta")
colnames(data) = c('countryCode', 'entryYear', 'countryName', 'avgTemp',
                   'avgPrecip', 'seaDist', 'avgElev', 'sdElev', 'absLat',
                   'dispElev', 'numLang', 'suitableCells', 'dispSuitable',
                   'climate', 'soil', 'sdClimate', 'sdSoil', 'sdSuitable',
                   'avgSuitable', 'pop95', 'area', 'lnLang', 'africa',
                   'europe', 'americas', 'lnPopDens1995', 'migrationDist',
                   'lnArea', 'pctIndigenous', 'lnPopDens1500',
                   'agriTran', 'asiaPac')
greeceCells = countries %>% filter(COUNTRY == 'Greece') %>%
  st_intersection(y = cells)
nepalCells = countries %>% filter(COUNTRY == 'Nepal') %>%
  st_intersection(y = cells)
plot(
  density(greeceCells$suit_new, kernel = "epanechnikov"),
  xlim = c(0, 1),
  xlab = 'Land quality per region',
  ylab = 'Density',
  main = '',
  lty = 2
)
lines(density(nepalCells$suit_new, kernel = "epanechnikov"), col = 'red')
legend(
  .1,
  3,
  legend = c(
    'Distribution of land quality in Greece',
    'Distribution of land quality in Nepal'
  ),
  col = c("black", "red"),
  lty = 2:1,
  cex = .75
)
count = function(x) {
  (sum( ~ is.na(x)))
}

sumTable <- data %>% select(
  c(
    'numLang',
    'sdElev',
    'sdSuitable',
    'avgElev',
    'avgSuitable',
    'absLat',
    'avgPrecip',
    'avgTemp',
```

```
    'lnArea',
    'seaDist',
    'migrationDist',
    'lnPopDens1995'
  )
) %>%
  summarise_each(
    funs(
      min = min,
      median = median,
      max = max,
      mean = mean,
      iqr = quantile(., 0.75) - quantile(., 0.25),
      sd = sd,
      n = sum(!is.na(.))
    )
  ) %>%
  gather(var, val) %>%
  separate(var, into = c("var", "stat"), sep = "_") %>%
  spread(var, val) %>% column_to_rownames(var = "stat") %>%
  select(
    c(
      'numLang',
      'sdElev',
      'sdSuitable',
      'avgElev',
      'avgSuitable',
      'absLat',
      'avgPrecip',
      'avgTemp',
      'lnArea',
      'seaDist',
      'migrationDist',
      'lnPopDens1995'
    )
  ) %>%
  mutate_if(is.numeric, ~ round(., 2)) %>% slice(5, 4, 2, 3, 7)
standardize = function(vec) {return ((vec - mean(vec, na.rm = TRUE)) / sd(vec, na.rm = TRUE))}

modelCols = c('entryYear', 'avgTemp', 'avgPrecip', 'seaDist', 'avgElev',
              'sdElev', 'absLat', 'numLang', 'dispSuitable', 'climate',
              'soil', 'sdClimate', 'sdSoil', 'sdSuitable', 'avgSuitable',
              'pop95', 'area', 'lnLang', 'lnPopDens1995', 'migrationDist',
              'lnArea', 'pctIndigenous', 'lnPopDens1500', 'agriTran',
              'americas', 'europe', 'africa', 'asiaPac')
# for (col in modelCols) {
#  data[,col] = standardize(data[,col])
# }
# NOTE: Easier to say what we're not standardizing; also for the robustness checks we need to keep a non-standa
dataStd = data %>% mutate(across(!countryName & !countryCode , standardize))
```

```
model1.1 = lm(lnLang ~ absLat, dataStd)
model1.2 = lm(lnLang ~ sdElev + sdSuitable + avgElev + avgSuitable + absLat, dataStd)
model1.3 = lm(lnLang ~ sdElev + sdSuitable + avgElev + avgSuitable + absLat
            + avgPrecip + avgTemp + lnArea + seaDist + migrationDist, dataStd)
model1.4 = lm(lnLang ~ sdElev + sdSuitable + avgElev + avgSuitable + absLat
            + avgPrecip + avgTemp + lnArea + seaDist + migrationDist + lnPopDens1995
            + africa + europe + americas + asiaPac, dataStd)
missingData = is.na(dataStd$agriTran) | is.na(dataStd$entryYear) | is.na(dataStd$lnPopDens1500)
for (col in modelCols) {
  dataStd[!missingData, col] = standardize(dataStd[!missingData,col])
}

model1.5 = lm(lnLang ~ sdElev + sdSuitable + avgElev + avgSuitable + absLat
            + avgPrecip + avgTemp + lnArea + seaDist + migrationDist + lnPopDens1995
            + lnPopDens1500 + entryYear + agriTran
            + africa + europe + americas + asiaPac, dataStd, na.action = na.exclude)
models = paste0("model1.", 1:5)
coefs = sapply(models, function(model) {coeftest(get(model), vcov = vcovHC(get(model), "HC1"))[, 1]}) %>%
  unlist() %>% data.frame()
coefs$model = substr(row.names(coefs), 1, 8)
coefs$column = substr(row.names(coefs), 10, nchar(row.names(coefs)))

ses = sapply(models, function(model) {coeftest(get(model), vcov = vcovHC(get(model), "HC1"))[, 2]}) %>%
  unlist() %>% data.frame()
ses$model = substr(row.names(ses), 1, 8)
ses$column = substr(row.names(ses), 10, nchar(row.names(ses)))

pvals = sapply(models, function(model) {coeftest(get(model), vcov = vcovHC(get(model), "HC1"))[, 4]}) %>%
  unlist() %>% data.frame()
pvals$model = substr(row.names(pvals), 1, 8)
pvals$column = substr(row.names(pvals), 10, nchar(row.names(pvals)))

order = c('sdElev', 'sdSuitable', 'avgElev', 'avgSuitable', 'absLat',
          'avgPrecip', 'avgTemp', 'lnArea', 'seaDist', 'migrationDist',
          'lnPopDens1995', 'lnPopDens1500', 'entryYear', 'agriTran')

pvalsPivoted = pvals %>% pivot_wider(names_from = "model", values_from = '.') %>%
  slice(match(order, column))

tbl1 = rbind(coefs %>% pivot_wider(names_from = "model", values_from = '.'),
             ses %>% pivot_wider(names_from = "model", values_from = '.'))
tbl1$stat = c(rep('Estimate', 19), rep('SE', 19))
indices = c(rbind(match(order, tbl1$column), match(order, tbl1$column) + 19))
tbl1 = tbl1 %>% slice(indices)
tbl1format = data.frame(tbl1)
for (model in models) {
  estimRows = !is.na(tbl1[, model]) & (tbl1$stat == 'Estimate')
  seRows = !is.na(tbl1[, model]) & (tbl1$stat == 'SE')

  tbl1format[estimRows, model] = sprintf(fmt = "%.3f", tbl1[estimRows, model] %>% unlist() %>% as.numeric())
```

```
  tbl1format[seRows, model] = paste0("(", sprintf(fmt = "%.3f", tbl1[seRows, model] %>% unlist() %>% as.numeric

  significant = rep(pvalsPivoted[, model] < .01, each = 2)
  significant[is.na(significant)] = FALSE
  tbl1format[estimRows, model] = cell_spec(tbl1format[estimRows, model], italic = significant[estimRows])
  tbl1format[seRows, model] = cell_spec(tbl1format[seRows, model], italic = significant[seRows])
}

tbl1format$name = c('Variation in elevation', NA, 'Variation in land quality', NA,
                    'Mean elevation', NA, 'Mean land quality', NA,
                    'Absolute latitude', NA, 'Mean precipitation', NA,
                    'Mean temperature', NA, 'Ln(Area)', NA,
                    'Distance from the sea', NA, 'Migratory distance from Ethiopia', NA,
                    'Ln(Population density in 1995)', NA, 'Ln(Population density in 1500)', NA,
                    'Year of independence', NA, 'Timing of transition to agriculture', NA
                    )

col.names = c("Variable", paste0("(", 1:5, ")"))

tbl1format %>% select(8, 2:6) %>%
  knitr::kable(toprule = '', bottomrule = '', booktabs = TRUE, linesep = c("", "\\addlinespace"),
               col.names = col.names, align = "r", escape = F,
               caption = 'Main specification for the cross-country analysis. Italics indicate significance at the
  row_spec(seq(2, 28, 2), font_size = 8)
tbl1info = data.frame(
  model = 1:5,
  cont = c("No", "No", "No", "No", "Yes"),
  nobs = sapply(models, function(model) { nobs(get(model))}),
  rsq = sapply(models, function(model) {
    formatC(summary(get(model))$r.squared, digits = 2, format = 'f')
    }), row.names = NULL
  )
tbl1info %>%
  knitr::kable(toprule = '', bottomrule = '', booktabs = TRUE,
               col.names = c('Model', 'Continental Indicators', 'Observations', '$R^2$'), escape = F,
               caption = "Information for each model in cross-country analysis.")
yhat = model1.5$residuals
y = model1.5$model$lnLang
plot(y, yhat, xlab = "", ylab = "", cex.axis = .75)
title(ylab = "Residuals", xlab = "Ln(Number of Languages)", mgp = c(2, .5, 0), cex.lab = .75)
residModel = lm(yhat ~ y)
abline(0, 0)
lines(y, y * residModel$coefficients['y'], col = 'red', type = 'l')
robust1.1 <- glm.nb(numLang ~ absLat + sdSuitable + sdElev + avgElev + avgSuitable + avgPrecip + avgTemp + lnAr
                    + seaDist + migrationDist + lnPopDens1995 + lnPopDens1500 + entryYear + agriTran +
                      africa + europe + americas + asiaPac
                    , data, na.action = na.exclude)
robust1.2 <- lm(lnLang ~ absLat + dispElev + dispSuitable + avgElev + avgSuitable + avgPrecip +  avgTemp +
                  lnArea +  seaDist + migrationDist +  lnPopDens1995 + lnPopDens1500 + entryYear + agriTran +
                  africa + europe + americas + asiaPac,
```

```
                data, na.action = na.exclude)

robust1.3 <- lm(lnLang ~ absLat + sdElev + sdClimate + avgElev + climate + avgPrecip +  avgTemp +
                lnArea +  seaDist + migrationDist +  lnPopDens1995 +  lnPopDens1500 + entryYear + agriTran +
                africa + europe + americas + asiaPac,
                data, na.action = na.exclude)

# NOTE: the conditional doesn't remove anything from the `data` df
data1.4 <- data[(data$suitableCells > 9) & (data$lnArea > -10), ]

robust1.4 <- lm(lnLang ~ absLat + sdElev + sdSoil + avgElev + soil + avgPrecip +  avgTemp +
                lnArea +  seaDist + migrationDist +  lnPopDens1995 +  lnPopDens1500 + entryYear + agriTran +
                africa + europe + americas + asiaPac,
                data1.4, na.action = na.exclude)

data2b <- read.dta13("data_raw/Table_3b.dta")

standardized <- list("lpd1500", "yrentry", "agritran", "elf", "elf3", "elf5", "elf7", "elf9",
                     "abs_lat", "sd_climsuit", "sd_emean", "emean", "mean_climsuit", "precav",
                     "tempav", "lnareakm2", "distc", "migdist", "lnpop95", "americas", "reg_eap",
                     "africa", "europe", "nmbr_climsuit")

notStdized <- names(data2b)[! names(data2b) %in%  standardized]
data2bStd <- data2b %>% mutate(across(! all_of(notStdized) , standardize))

robust1.2.1 <- lm(elf ~ abs_lat, data2bStd, na.action = na.exclude)

robust1.2.2 <- lm(elf ~ abs_lat + sd_emean + sd_climsuit + emean + mean_climsuit,
                  data2bStd, na.action = na.exclude)

robust1.2.3 <- lm(elf ~ abs_lat + sd_emean + sd_climsuit + emean + mean_climsuit +
                  precav + tempav + lnareakm2 + distc + migdist + lnpop95 + lpd1500 + yrentry + agritran +
                  africa + europe + americas + reg_eap
                  , data2bStd, na.action = na.exclude)

robust1.2.4 <- lm(elf3 ~ abs_lat + sd_emean + sd_climsuit + emean + mean_climsuit +
                  precav + tempav + lnareakm2 + distc + migdist + lnpop95 + lpd1500 + yrentry + agritran +
                  africa + europe + americas + reg_eap
                  , data2bStd, na.action = na.exclude)

robust1.2.5 <- lm(elf5 ~ abs_lat + sd_emean + sd_climsuit + emean + mean_climsuit +
                  precav + tempav + lnareakm2 + distc + migdist + lnpop95 + lpd1500 + yrentry + agritran +
                  africa + europe + americas + reg_eap
                  , data2bStd, na.action = na.exclude)

robust1.2.6 <- lm(elf7 ~ abs_lat + sd_emean + sd_climsuit + emean + mean_climsuit +
                  precav + tempav + lnareakm2 + distc + migdist + lnpop95 + lpd1500 + yrentry + agritran +
                  africa + europe + americas + reg_eap
                  , data2bStd, na.action = na.exclude)
```

```
robust1.2.7 <- lm(elf9 ~ abs_lat + sd_emean + sd_climsuit + emean + mean_climsuit +
                  precav + tempav + lnareakm2 + distc + migdist + lnpop95 + lpd1500 + yrentry + agritran +
                  africa + europe + americas + reg_eap
                  , data2bStd, na.action = na.exclude)

data2 = read.dta13('data_raw/Tables4-7b.dta')
greg = read.csv('greg.csv')
colnames(greg) = c('uniq_cnt25', 'number_suit_valid25', 'nmbrlang')

data2 = data2 %>% select(-c('nmbrlang', 'number_suit_valid25')) %>% merge(greg, by = 'uniq_cnt25')

data2$lnnmbrlang = log(data2$nmbrlang)
colnames(data2) = c('virtCode', 'countryCode', 'climate', 'soil',
                    'sdClimate', 'sdSoil', 'seaDist', 'avgElev', 'avgPrecip',
                    'avgTemp', 'sdElev', 'waterArea', 'avgSuitable',
                    'sdSuitable', 'popDens95', 'range', 'area', 'withinCountry',
                    'numCountry', 'migrationDist', 'lnLang', 'totalPop95',
                    'absLat', 'tropics', 'erange_gecon', 'lnArea',
                    'lnPopDens95', 'pctIndigenous', 'diffAvgElev',
                    'diffAvgPrecip', 'diffAvgTemp', 'diffAvgSuit',
                    'overlap', 'suitableCells', 'numLang')

modelCols = c('lnLang', 'sdElev', 'sdSuitable', 'avgElev', 'avgSuitable',
              'absLat', 'avgPrecip', 'avgTemp', 'lnArea', 'seaDist', 'waterArea',
              'withinCountry', 'numCountry', 'migrationDist', 'lnPopDens95')
condition = (data2$totalPop95 >= 3000) & (data2$suitableCells >= 10)
for (col in modelCols) {
  data2[condition, paste0(col, '1')] = standardize(data2[condition, col])
}
model2.1 = lm(lnLang1 ~ absLat1, data2 %>% filter(condition))
coefs = coeftest(model2.1, vcov = vcovCL, cluster = ~countryCode)[, 1]
ses = coeftest(model2.1, vcov = vcovCL, cluster = ~countryCode)[, 2]
pvals = coeftest(model2.1, vcov = vcovCL, cluster = ~countryCode)[, 4]
model2.2 = lm(lnLang1 ~ sdElev1 + sdSuitable1 + avgElev1 + avgSuitable1 + absLat1,
              data2 %>% filter(condition))
coefs = c(coefs, coeftest(model2.2, vcov = vcovCL, cluster = ~countryCode)[, 1])
ses = c(ses, coeftest(model2.2, vcov = vcovCL, cluster = ~countryCode)[, 2])
pvals = c(pvals, coeftest(model2.2, vcov = vcovCL, cluster = ~countryCode)[, 4])
model2.3 = lm(lnLang1 ~ sdElev1 + sdSuitable1 + avgElev1 + avgSuitable1 + absLat1
              + avgPrecip1 + avgTemp1 + lnArea1 + seaDist1 + waterArea1
              + withinCountry1 + numCountry1 + migrationDist1,
              data2 %>% filter(condition))
coefs = c(coefs, coeftest(model2.3, vcov = vcovCL, cluster = ~countryCode)[, 1])
ses = c(ses, coeftest(model2.3, vcov = vcovCL, cluster = ~countryCode)[, 2])
pvals = c(pvals, coeftest(model2.3, vcov = vcovCL, cluster = ~countryCode)[, 4])
model2.4 = lm_robust(lnLang1 ~ sdElev1 + sdSuitable1 + avgElev1 + avgSuitable1 + absLat1
              + avgPrecip1 + avgTemp1 + lnArea1 + seaDist1 + waterArea1
              + withinCountry1 + numCountry1 + migrationDist1 + lnPopDens951,
              data2 %>% filter(condition),
              fixed_effects = ~countryCode, se_type = "stata")
```

```
coefs = c(coefs, 0, model2.4$coefficients)
ses = c(ses, 0, model2.4$std.error)
pvals = c(pvals, 0, model2.4$p.value)
condition = (data2$totalPop95 >= 3000) & (data2$suitableCells >= 10) & (data2$tropics == 1)
for (col in modelCols) {
  data2[condition, paste0(col, '5')] = standardize(data2[condition, col])
}
model2.5 = lm_robust(lnLang5 ~ sdElev5 + sdSuitable5 + avgElev5 + avgSuitable5 + absLat5
            + avgPrecip5 + avgTemp5 + lnArea5 + seaDist5 + waterArea5
            + withinCountry5 + numCountry5 + migrationDist5 + lnPopDens955,
            data2 %>% filter(condition),
            fixed_effects = ~countryCode, se_type = "stata")
coefs = c(coefs, 0, model2.5$coefficients)
ses = c(ses, 0, model2.5$std.error)
pvals = c(pvals, 0, model2.5$p.value)
condition = (data2$totalPop95 >= 3000) & (data2$suitableCells >= 10) & (data2$tropics == 0)
for (col in modelCols) {
  data2[condition, paste0(col, '6')] = standardize(data2[condition, col])
}
model2.6 = lm_robust(lnLang6 ~ sdElev6 + sdSuitable6 + avgElev6 + avgSuitable6 + absLat6
            + avgPrecip6 + avgTemp6 + lnArea6 + seaDist6 + waterArea6
            + withinCountry6 + numCountry6 + migrationDist6 + lnPopDens956,
            data2 %>% filter(condition),
            fixed_effects = ~countryCode, se_type = "stata")
coefs = c(coefs, 0, model2.6$coefficients)
ses = c(ses, 0, model2.6$std.error)
pvals = c(pvals, 0, model2.6$p.value)
condition = (data2$totalPop95 >= 3000) & (data2$suitableCells >= 10) & (data2$withinCountry == 1)
for (col in modelCols) {
  data2[condition, paste0(col, '7')] = standardize(data2[condition, col])
}
model2.7 = lm_robust(lnLang7 ~ sdElev7 + sdSuitable7 + avgElev7 + avgSuitable7 + absLat7
            + avgPrecip7 + avgTemp7 + lnArea7 + seaDist7 + waterArea7
            + migrationDist7 + lnPopDens957,
            data2 %>% filter(condition),
            fixed_effects = ~countryCode, se_type = "stata")
coefs = c(coefs, 0, model2.7$coefficients)
ses = c(ses, 0, model2.7$std.error)
pvals = c(pvals, 0, model2.7$p.value)
models = paste0("model2.", 1:7)

names(coefs)[names(coefs) == ""] = "(Intercept)"
coefs = data.frame(coefs, row.names = paste0("model2.", cumsum(names(coefs) %in% c("(Intercept)", "")), ".", na
coefs$model = substr(row.names(coefs), 1, 8)
coefs$column = substr(row.names(coefs), 10, nchar(row.names(coefs)) - 1)

names(ses)[names(ses) == ""] = "(Intercept)"
ses = data.frame(ses, row.names = paste0("model2.", cumsum(names(ses) %in% c("(Intercept)", "")), ".", names(se
ses$model = substr(row.names(ses), 1, 8)
ses$column = substr(row.names(ses), 10, nchar(row.names(ses)) - 1)
```

```
names(pvals)[names(pvals) == ""] = "(Intercept)"
pvals = data.frame(pvals, row.names = paste0("model2.", cumsum(names(pvals) %in% c("(Intercept)", "")), ".", na
pvals$model = substr(row.names(pvals), 1, 8)
pvals$column = substr(row.names(pvals), 10, nchar(row.names(pvals)) - 1)

order = c('sdElev', 'sdSuitable', 'avgElev', 'avgSuitable', 'absLat',
          'avgPrecip', 'avgTemp', 'lnArea', 'seaDist', 'waterArea',
          'withinCountry', 'numCountry', 'migrationDist', 'lnPopDens95')

pvalsPivoted = pvals %>% pivot_wider(names_from = "model", values_from = 'pvals') %>%
  slice(match(order, column))

tbl4 = rbind(coefs %>% pivot_wider(names_from = "model", values_from = 'coefs'),
             ses %>% pivot_wider(names_from = "model", values_from = 'ses'))
tbl4$stat = c(rep('Estimate', 15), rep('SE', 15))
indices = c(rbind(match(order, tbl4$column), match(order, tbl4$column) + 15))
tbl4 = tbl4 %>% slice(indices)
tbl4format = data.frame(tbl4)
for (model in models) {
  estimRows = !is.na(tbl4[, model]) & (tbl4$stat == 'Estimate')
  seRows = !is.na(tbl4[, model]) & (tbl4$stat == 'SE')

  tbl4format[estimRows, model] = sprintf(fmt = "%.3f", tbl4[estimRows, model] %>% unlist() %>% as.numeric())
  tbl4format[seRows, model] = paste0("(", sprintf(fmt = "%.3f", tbl4[seRows, model] %>% unlist() %>% as.numeric

  significant = rep(pvalsPivoted[, model] < .01, each = 2)
  significant[is.na(significant)] = FALSE
  tbl4format[estimRows, model] = cell_spec(tbl4format[estimRows, model], italic = significant[estimRows])
  tbl4format[seRows, model] = cell_spec(tbl4format[seRows, model], italic = significant[seRows])
}

tbl4format$name = c('Variation in elevation', NA, 'Variation in land quality', NA,
                    'Mean elevation', NA, 'Mean land quality', NA,
                    'Absolute latitude', NA, 'Mean precipitation', NA,
                    'Mean temperature', NA, 'Ln(Area)', NA,
                    'Distance from the sea', NA, 'Water area', NA,
                   'Within-country indicator', NA, 'Number of countries', NA,
                   'Migratory distance from Ethiopia', NA, 'Ln(Population density in 1995)', NA)

col.names = c("Variable", paste0("(", 1:7, ")"))

tbl4format %>% select(10, 2:8) %>%
  knitr::kable(toprule = '', bottomrule = '', booktabs = TRUE, linesep = c("", "\\addlinespace"),
               col.names = col.names, align = "r", escape = F,
               caption = 'Main specification for the virtual country analysis. Italics indicate significance at t
  row_spec(seq(2, 28, 2), font_size = 8)
tbl4info = data.frame(
  model = 1:7,
  cont = c("No", "No", "No", "Yes", "Yes", "Yes", "Yes"),
  nobs = sapply(models, function(model) { nobs(get(model))}),
```

```
  rsqOG = formatC(c(.31, .36, .53, .70, .73, .56, .66), digits = 2, format = 'f'),
  rsq = sapply(models, function(model) {
    formatC(summary(get(model))$r.squared, digits = 2, format = 'f')
    }),
  row.names = NULL
  )
tbl4info %>%
  knitr::kable(toprule = '', bottomrule = '', booktabs = TRUE,
               linesep = c(""), align = 'r',
               col.names = c('Model', 'Country Indicators', 'Observations', 'WLMS $R^2$', 'GREG $R^2$'), escape
               caption = "Information for each model in virtual country analysis.")
```