Predicting semantic alignment by cultural similarity

Bill Thompson, Seán Roberts & Gary Lupyan

Contents

Introduction

We compare cultural distances between socieites with semantic alignment between societies, controlling for shared history in two ways.

The first test uses mixed effects modelling. The pairing of the language family of each language (according to Glottolog, Hammarstrom et al., 2018) is used as a random effect. That means that the model can capture the likelihood that two languages from the same language family (e.g. Indo-European) will be more similar to each other than two languages from different language families. The same is done with geographic area according to Autotyp (Nichols et al., 2013), which reflect areas of known linguistic contact. The model also included a fixed effect for for the number of lexical comparisons that went into the mean semantic alignment estimates (generally, more available comparisons indicate more possible comparisons, i.e. more similar languages).

The second and thrid test controls for history using distances from a phylogenetic tree. The tree comes from Bouckaert et al. (2012), which was estimated by comapring the gain and loss of cognates in the lexicon of Indo-European languages. Patristic distances between languages are used as a measure of historical distance between languages. Patristic distance is the distance between two leaves on the tree following the shortest path (which will go through the most recent common ancestor). The branch lengths in the tree are scaled to reflect time, so the shortest distance between two leaves on the tree indicates the total amount of independent evolution between two languages. An alternative measure of historical distance was obtained from data from the Automated Similarity Judgement Program database (ASJP, Wichmann, Holman & Brown, 2018). This is a databse of basic vocabulary in a common phonetic format. We used the distances as calculated by Jäger (2018) which essentially measure the average edit distance between langauges (the number of changes to turn one vocabulary into the other), accounting for the likelihood of historical changes between sound segments. The measure is similar to calculating distances between sequences of DNA. Both of these measures of historical distance are based on the lexicon, but do not use measures of the semantic meanings of words.

The second test uses simple and partial Mantel tests (Mantel, 1967 and e.g. Smouse, Long & Sokal, 1986, Legendre, 2000; Castellano & Balletto, 2002, Goslee, 2010), using the implementation in the R package *ecodist* (Goslee & Urban, 2007). A Mantel test is a nonparemetric test that uses permutation to assess the strength of the relationship between two distance matrices. It compares the correlation between the values from two distance matrices with the correlation produced when the values of one of the matrices is permuted. This allows it to account for the dependencies between the distances. Note that the Mantel test assumes a strict distance metric, which is not necessarily the case with this data (see also e.g. Harmon & Glor, 2010), but there are few other ways to deal with continuous pairwise distances.

Mantel tests were used to test the relationship between semantic alignment and historical proximity, and between semantic alignment and geographic proximity. Geographic proximity was measured as the great circle distance between the cultural centres of each language as defined in Glottolog (Hammarstrom et al., 2018). For the analysis within domains, partial Mantel tests were used to estimate the correlation between the semantic alignment and the cultural/geographic/historical proximity while partialling out the effect of the other two proximity measures.

The third test uses multiple regression on distance matrices (Lichstein, 2007). This is a regression approach which uses distance matrices as dependent and independent variables.

The results above were robustly replicated using the filtered data and also alternative sources for semantic alignment (common crawl, see file *AnalyseCorrelation_cc.pdf*). The correlation was not robust to all tests or for data derived from the subtitles dataset (see file *AnalyseCorrelation_subs.pdf*), possibly because there were only 20 languages available to analyse.

The final section looks at relationships between sub-domains. The first section describes how the cultural similarity measure was calculated.

Calculating cultural similarity

The aim is to produce a set of distances between societies based on their cultural traits. The Ethnographic Atlas (Murdock et al., 1999) is a database of (non-linguistic) cultural traits on many societies. For each variable, societies are assigned to one category (or value). For example, the variable 'EA011' classifies a society's norms for "Transfer of residence at marriage". Each society is assigned to one of the following groups: "Wife to husband's group", "Husband to wife's group" "Couple to either group", "Nonestablishment of a common household". The D-PLACE database (https://d-place.org/, Kirby et al., 2016) links societies in the Ethnogrpahic atlas to the languages they speak (through the Glottolog ID, Hammarstrom et al., 2018). D-PLACE also provides the data in an updated format, so we use this as our primary data source.

However, there is a lot of missing data in the Ethnographic Atlas (about 25% in the whole dataset), which means that distances can't be computed easily. One approach is to impute the missing data (guess their values based on existing data). It's unlikely that any imputation method will be completely accurate, but for our purposes we don't need to be accurate, just *unbiased*. That is, the imputed values should not bias the estimates of the distances between cultures.

In this case, we use multiple imputation: calculating many possible alternative imputations and taking the mean distances over all imputations.

Imputing missing values in the Ethnographic Atlas

We use the imputation package `mice` for R (van Buuren & Groothuis-Oudshoorn, 2011). We compared various settings of the imputation method, and found that using classification and regression trees (CART) with the standard parameters produced the best results. CART works by building a decision tree: an optimal set of yes-no questions to ask about predictor variables in order to guess the value of a target variable. The tree divides the data into partitions which look similar. The algorithm works out which partition a missing data point would belong to, then samples the target variable distribution from that partition. To account for historical relationships, we included language family according to Glottolog and geographic area according to Autotyp as additional factors on which the imputation process could draw.

We ran CART multiple imputation on the Ethnographic Atlas. We excluded population size, one more variable that was coded for less than 33% of societies, and any societies that had fewer than 33% variables coded. This left 92 variables for 962 languages with 16% missing data.

We tested the imputation by taking the full Ethnographic Atlas data, creating some new missing values in random places and then re-imputing those missing values. We can then asses how accurate the imputation was for those values. Since the main analysis would only be using a small sub-set of the data, it is important to assess performance on these in particular, rather than the entire set of languages. Missing data was only inserted for languages in the main analysis of semantic alignment below. CART imputation guessed the correct value of missing data 74% of the time (average over 100 imputations). This is reasonably good, considering that most variables have between 4 and 8 possible values (median = 6). For example, this is 8.6 standard deviations better than choosing randomly (accuracy = 19%) and 5.6 standard deviations better than sampling from the known distribution of the target variable (accuracy = 37% on the same missing data). This is not good enough to use in analyses that look at individual traits, but serves our purposes to estimate overall distances between langauges.

We produced 100 imputation sets with the final settings. These were then used to create distance matrices using Gower distance between discrete traits (mean correlation between sets r = 0.94, estimates of distance vary by around 2% on average). The final distance matrix was the mean of each of the 100 distance matrices. Distances were also calculate for sub-domains of the data.

The full scripts and data can be found at https://github.com/seannyD/ImputeEACulturalDifferences. Reviewers can follow this link: https://figshare.com/s/06378bc59a771d28b1d0

Load libraries

```
library(ape)
library(ecodist)
library(lme4)
library(sjPlot)
library(ggplot2)
library(igraph)
library(lattice)
library(dplyr)
```

Parameters (using data from Northuralex and Wikipedia, k=100, unfiltered):

```
datasetName = "wikipedia-main"
lingDistancesFile = "../data/FAIR/nel-wiki-k100-alignments-by-language-pair.csv"
lingDistancesFileNK = "../data/FAIR/nel-wiki-k100-alignments-by-language-pair-without-kinship.csv"
lingDistancesByDomainFile = "../results/EA_distances/nel-wiki-k100_with_ling.csv"
# (generated by ../processing/combineCultAndLingDistances.R)
```

All domains

Load data

Read the cultural distances:

```
cult = read.csv("../results/EA_distances/CulturalDistances_Long.csv", stringsAsFactors = F)
names(cult) = c("l1","l2","cult.dist")
```

Add language family:

```
l = read.csv("../data/FAIR_langauges_glotto_xdid.csv", stringsAsFactors = F)
g = read.csv("../data/glottolog-languoid.csv/languoid.csv", stringsAsFactors = F)
l$family = g[match(l$glotto,g$id),]$family_pk
l$family = g[match(l$family,g$pk),]$name
```

Read the semantic distances

```
ling = read.csv(lingDistancesFile, stringsAsFactors = F)
```

There are very few possible comparisons for Slovenian and Northern Sami, so we'll remove these:

```
ling = ling[!(ling$l1=="se" | ling$l2 == "se"),]
ling = ling[!(ling$l1=="sl" | ling$l2 == "sl"),]
```

Combine the lingusitic and cultural distances. Note that we flip the cultural measure from a distance measure to a similarity measure.

```
cult$l1.iso2 = l[match(cult$l1,l$Language2),]$iso2
cult$l2.iso2 = l[match(cult$l2,l$Language2),]$iso2

fairisos = unique(c(ling$l1,ling$l2))
cultisos = unique(c(cult$l1.iso2, cult$l2.iso2))

cult = cult[(cult$l1.iso2 %in% fairisos) & (cult$l2.iso2 %in% fairisos),]
ling = ling[(ling$l1 %in% cultisos) & (ling$l2 %in% cultisos),]

matches = sapply(1:nrow(ling), function(i){
  which(cult$l1.iso2==ling$l1[i] & cult$l2.iso2==ling$l2[i])
})

ling$cult.dist = cult[matches,]$cult.dist
# Flip
ling$cult.dist = 1 - ling$cult.dist
# Scale
ling$cult.dist.center = scale(ling$cult.dist)
cdc.s = attr(ling$cult.dist.center,"scaled:scale")
cdc.c = attr(ling$cult.dist.center,"scaled:center")
ling$cult.dist.center = as.numeric(ling$cult.dist.center)
ling$comparison_count.center =
  scale(ling$comparison_count)

ling$family1 = l[match(ling$l1, l$iso2),]$family
ling$family2 = l[match(ling$l2, l$iso2),]$family
l[l$Language=="Arabic",]$autotyp.area= "Greater Mesopotamia"
l[l$Language=="Persian",]$autotyp.area= "Greater Mesopotamia"
ling$area1 = l[match(ling$l1, l$iso2),]$autotyp.area
```

```
ling$area2 = l[match(ling$l2, l$iso2),]$autotyp.area


fgroup = cbind(ling$family1,ling$family2)
fgroup = apply(fgroup,1,sort)
ling$family.group = apply(fgroup,2,paste,collapse=":")
agroup = cbind(ling$area1,ling$area2)
agroup = apply(agroup,1,sort)
ling$area.group = apply(agroup,2,paste,collapse=":")

ling$rho.center = scale(ling$local_alignment)
```

Each observation is now assocaited with a language family pair:

```
head(ling[,c("l1","l2","local_alignment",'family.group')])
```

```
##     l1  l2 local_alignment               family.group
## 7   ja  ab      0.01930414       Abkhaz-Adyge:Japonic
## 8   ab  zh      0.02225169  Abkhaz-Adyge:Sino-Tibetan
## 10  cv xal      0.02765860            Mongolic:Turkic
## 11 xal  ja      0.02832668           Japonic:Mongolic
## 12 xal  zh      0.02895876      Mongolic:Sino-Tibetan
## 14  bn  ab      0.03192066 Abkhaz-Adyge:Indo-European
```

And the same is true for area:

```
tail(ling[,c("l1","l2","local_alignment",'area.group')])
```

```
##      l1 l2 local_alignment            area.group
## 2522 fr es       0.3936442         Europe:Europe
## 2524 cs uk       0.4023323     Europe:Inner Asia
## 2528 cs ru       0.4082099     Europe:Inner Asia
## 2529 be ru       0.4129814 Inner Asia:Inner Asia
## 2532 uk be       0.4276664 Inner Asia:Inner Asia
## 2535 uk ru       0.5079911 Inner Asia:Inner Asia
```

Number of observations:

```
# Number of datapoints:
nrow(ling)
```

```
## [1] 731
```

```
# Number of unique languages:
length(unique(unlist(ling[,c("l1","l2")])))
```

```
## [1] 39
```

```
# Number of unique langauge families:
uniqueFamilies = unique(unlist(ling[,c("family1","family2")]))
length(uniqueFamilies)
```

```
## [1] 10
```

```
# Number of unique areas:
uniqueAreas = unique(unlist(ling[,c("area1","area2")]))
length(uniqueAreas)
```

```
## [1] 6
```

Cross-over between language famlies and areas:

```
tx = data.frame(lang= c(ling$l1,ling$l2),
           fam = c(ling$family1,ling$family2),
           area= c(ling$area1,ling$area2))
tx = tx[!duplicated(tx),]
table(tx$fam,tx$area)
```

```
##
##                 Europe Greater Mesopotamia Indic Inner Asia N Coast Asia
##   Abkhaz-Adyge       0                   1     0          0            0
##   Afro-Asiatic       0                   1     0          0            0
##   Dravidian          0                   0     3          0            0
##   Indo-European     11                   2     1          5            0
##   Japonic            0                   0     0          0            1
##   Koreanic           0                   0     0          0            1
##   Mongolic           0                   0     0          1            0
##   Sino-Tibetan       0                   0     0          0            0
##   Turkic             0                   1     0          5            0
##   Uralic             1                   0     0          4            0
##
##                 Southeast Asia
##   Abkhaz-Adyge                0
##   Afro-Asiatic                0
##   Dravidian                   0
##   Indo-European               0
##   Japonic                     0
##   Koreanic                    0
##   Mongolic                    0
##   Sino-Tibetan                1
##   Turkic                      0
##   Uralic                      0
```

LMER models

Mixed effects model, predicting semantic alignment from cultural similarity, with random intercept for family and area and random slope for cultural similarity for family and area.

We start with a null model with random intercepts for family and area, and random slopes for cultural similarity by both. We add a fixed effect of the number of comparisons made for each datapoint (number of concepts that were available to compare). Then we add a fixed effect of cultural similarity

```
m0 = lmer(
  rho.center ~ 1 +
    (1 + cult.dist.center | family.group) +
    (1 + cult.dist.center | area.group),
  data = ling
)
```

```
## boundary (singular) fit: see ?isSingular
```

```
m0.5 = lmer(
  rho.center ~ 1 +
    comparison_count.center +
    (1 + cult.dist.center | family.group) +
    (1 + cult.dist.center | area.group),
  data = ling
)
```

```
## boundary (singular) fit: see ?isSingular
```

```
m1 = lmer(
  rho.center ~ 1 +
    comparison_count.center +
    cult.dist.center +
    (1 + cult.dist.center | family.group) +
    (1 + cult.dist.center | area.group),
  data = ling
)
```

```
## boundary (singular) fit: see ?isSingular
```

```
an1 = anova(m0,m0.5,m1)
```

```
## refitting model(s) with ML (instead of REML)
```

```
an1
```

```
## Data: ling
## Models:
## m0: rho.center ~ 1 + (1 + cult.dist.center | family.group) + (1 +
## m0:     cult.dist.center | area.group)
## m0.5: rho.center ~ 1 + comparison_count.center + (1 + cult.dist.center |
## m0.5:     family.group) + (1 + cult.dist.center | area.group)
## m1: rho.center ~ 1 + comparison_count.center + cult.dist.center +
## m1:     (1 + cult.dist.center | family.group) + (1 + cult.dist.center |
## m1:     area.group)
##      Df    AIC    BIC  logLik deviance   Chisq Chi Df Pr(>Chisq)
## m0    8 1654.6 1691.3 -819.30   1638.6
## m0.5  9 1293.0 1334.3 -637.50   1275.0 363.597      1  < 2.2e-16 ***
## m1   10 1278.4 1324.4 -629.22   1258.4  16.564      1  4.704e-05 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Cultural similarity is not significantly correlated with semantic alignment. Here are the model estimates:

```
summary(m1)
```

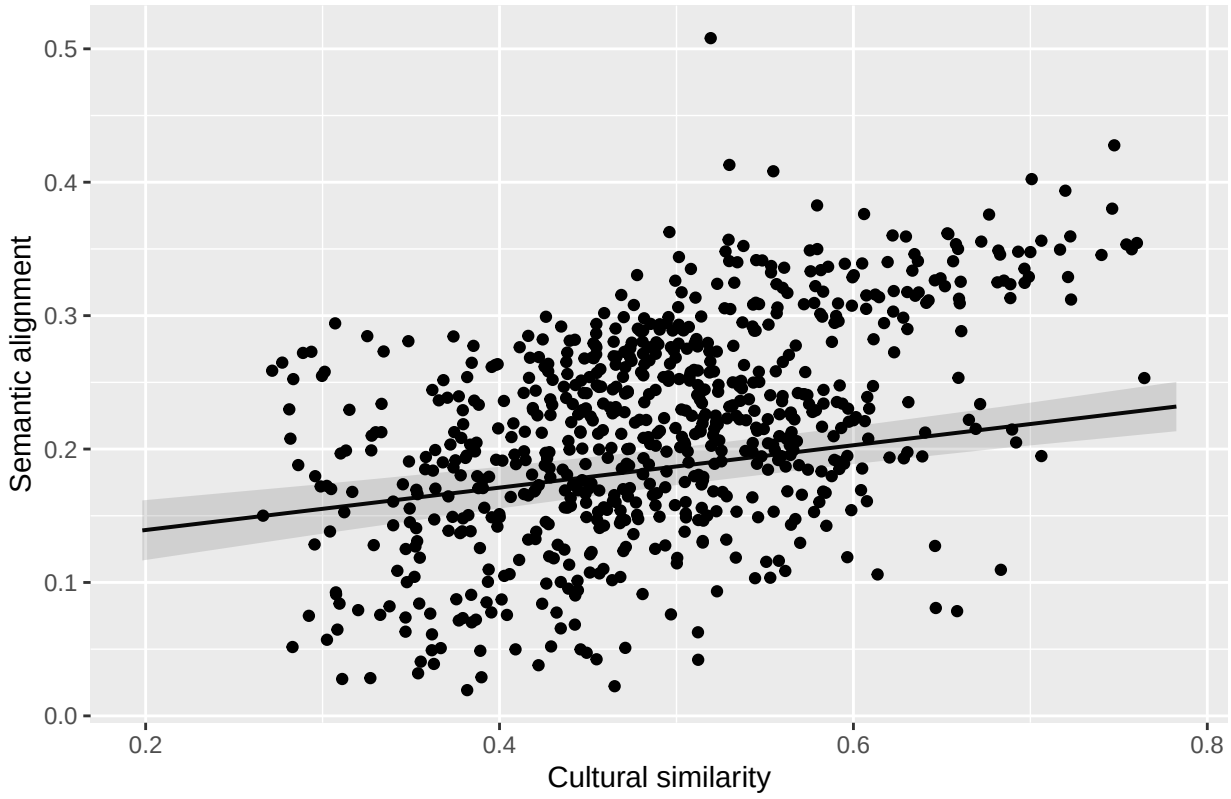
```
## Linear mixed model fit by REML ['lmerMod']
## Formula: rho.center ~ 1 + comparison_count.center + cult.dist.center +
##     (1 + cult.dist.center | family.group) + (1 + cult.dist.center |
##     area.group)
##    Data: ling
##
## REML criterion at convergence: 1271.9
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.6249 -0.6167  0.1103  0.6571  4.7402
##
## Random effects:
##  Groups       Name             Variance  Std.Dev. Corr
##  family.group (Intercept)      0.1612630 0.40158
##               cult.dist.center 0.0001817 0.01348  1.00
##  area.group   (Intercept)      0.0510850 0.22602
##               cult.dist.center 0.0036658 0.06055  -1.00
##  Residual                      0.2885416 0.53716
## Number of obs: 731, groups:  family.group, 48; area.group, 20
##
## Fixed effects:
##                          Estimate Std. Error t value
## (Intercept)             -0.39270    0.09073  -4.328
## comparison_count.center  0.61196    0.02688  22.770
## cult.dist.center         0.19678    0.03275   6.008
##
## Correlation of Fixed Effects:
##             (Intr) cmpr_.
## cmprsn_cnt.  0.090
## clt.dst.cnt -0.194 -0.201
## convergence code: 0
## boundary (singular) fit: see ?isSingular
```

Plot the estimates, rescaling the variables back to the original units:

```
trans = function(X){
  X * attr(ling$rho.center,"scaled:scale") +
  attr(ling$rho.center,"scaled:center")
}

gx = plot_model(m1,'pred',terms='cult.dist.center')
gx$data$predicted = trans(gx$data$predicted)
gx$data$conf.low = trans(gx$data$conf.low)
gx$data$conf.high = trans(gx$data$conf.high)
gx$data$x = gx$data$x *
  cdc.s +cdc.c
gx = gx + #coord_cartesian(ylim=c(0,0.5),
          #                xlim=c(0.15,0.85)) +
  xlab("Cultural similarity") +
```

```
  ylab("Semantic alignment") +
  ggtitle("") +
  geom_point(data=ling,aes(x=cult.dist,y=local_alignment))
gx
```



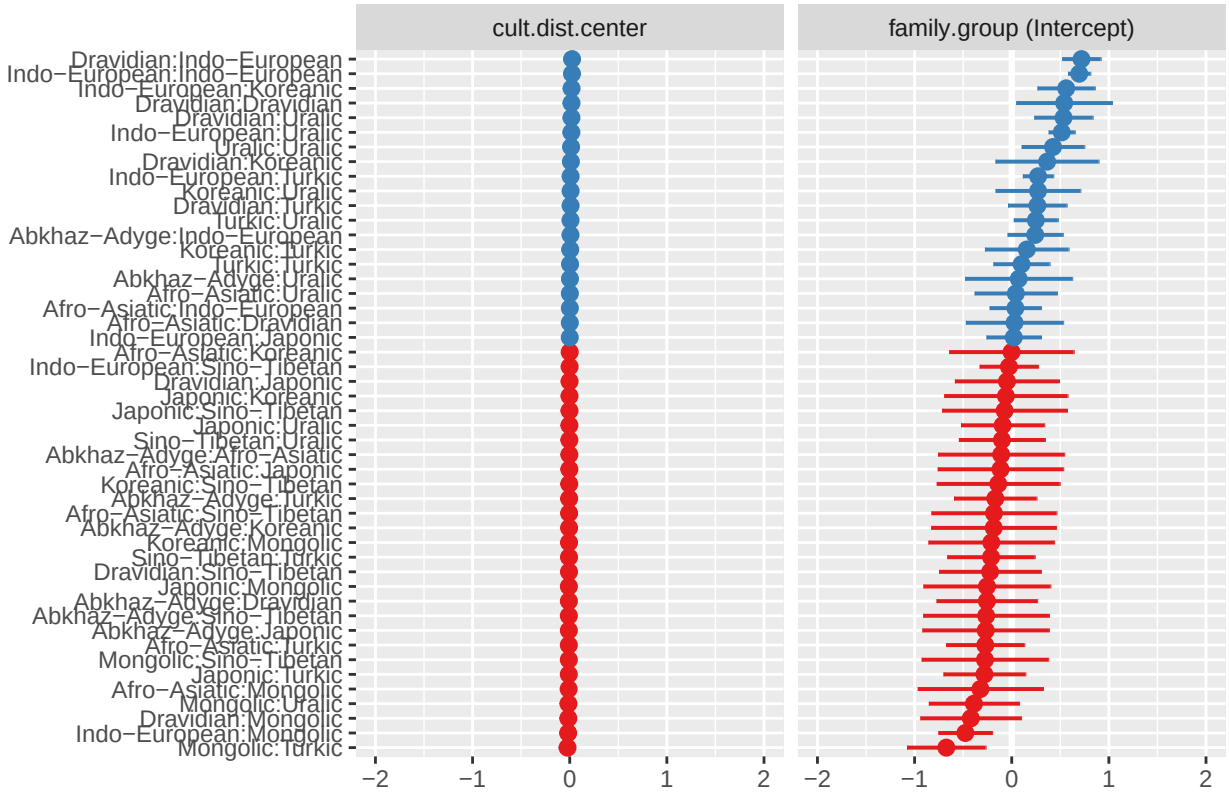

```
pdf(paste0("../results/stats/",datasetName,"/CulturalDistance_Rho_Graph.pdf"),
    height=2.5, width=2.5)
gx
dev.off()
```

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

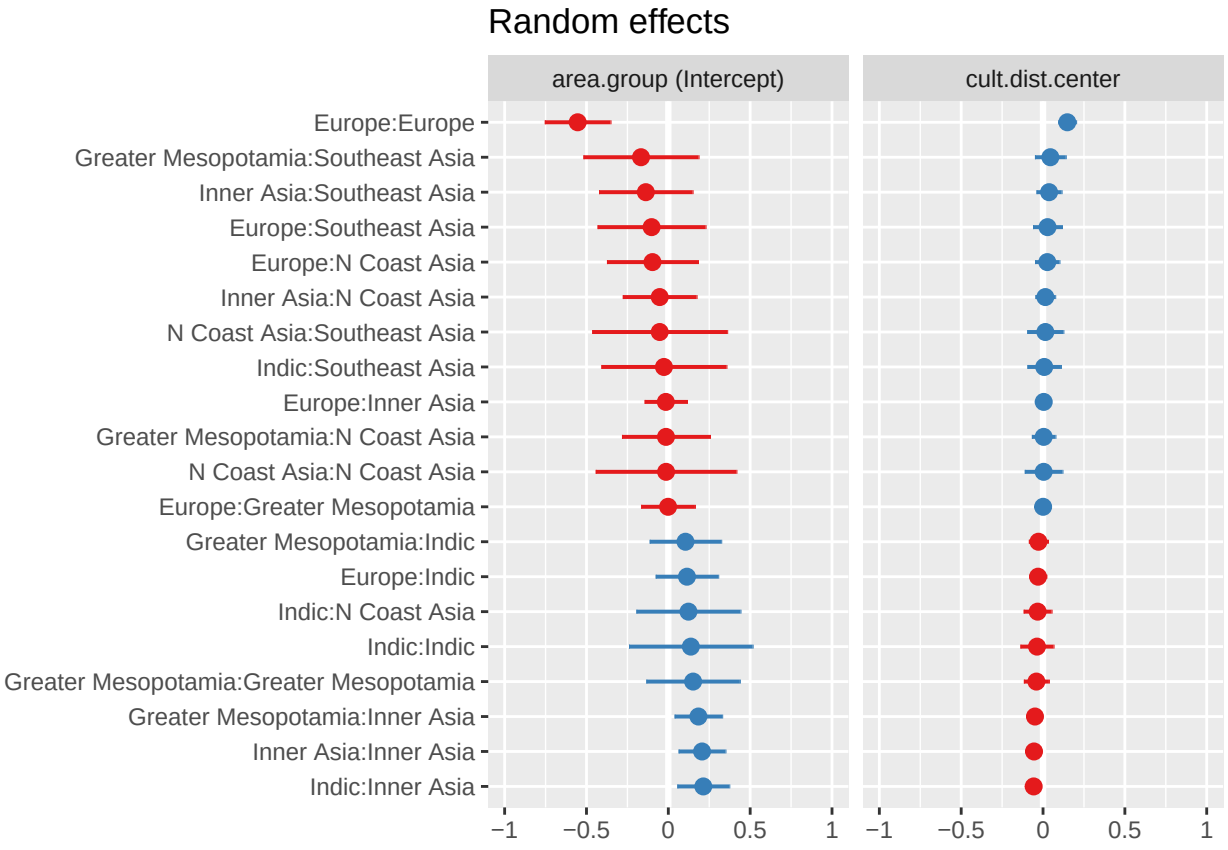
Plot the random effects:

```
plot_model(m1,'re', sort.est = "cult.dist.center")
```

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


Random effects
cult.dist.center
family.group (Intercept)
Dravidian:Indo−European
Indo−European:Indo−European
Indo−European:Koreanic
Dravidian:Dravidian
Dravidian:Uralic
Indo−European:Uralic
Uralic:Uralic
Dravidian:Koreanic
Indo−European:Turkic
Koreanic:Uralic
Dravidian:Turkic
Turkic:Uralic
Abkhaz−Adyge:Indo−European
Koreanic:Turkic
Turkic:Turkic
Abkhaz−Adyge:Uralic
Afro−Asiatic:Uralic
Afro−Asiatic:Indo−European
Afro−Asiatic:Dravidian
Indo−European:Japonic
Afro−Asiatic:Koreanic
Indo−European:Sino−Tibetan
Dravidian:Japonic
Japonic:Koreanic
Japonic:Sino−Tibetan
Japonic:Uralic
Sino−Tibetan:Uralic
Abkhaz−Adyge:Afro−Asiatic
Afro−Asiatic:Japonic
Koreanic:Sino−Tibetan
Abkhaz−Adyge:Turkic
Afro−Asiatic:Sino−Tibetan
Abkhaz−Adyge:Koreanic
Koreanic:Mongolic
Sino−Tibetan:Turkic
Dravidian:Sino−Tibetan
Japonic:Mongolic
Abkhaz−Adyge:Dravidian
Abkhaz−Adyge:Sino−Tibetan
Abkhaz−Adyge:Japonic
Afro−Asiatic:Turkic
Mongolic:Sino−Tibetan
Japonic:Turkic
Afro−Asiatic:Mongolic
Mongolic:Uralic
Dravidian:Mongolic
Indo−European:Mongolic
Mongolic:Turkic
−2
−1
0
1
2

```
##
## [[2]]
```


Random effects
area.group (Intercept)
cult.dist.center
Europe:Europe
Greater Mesopotamia:Southeast Asia
Inner Asia:Southeast Asia
Europe:Southeast Asia
Europe:N Coast Asia
Inner Asia:N Coast Asia
N Coast Asia:Southeast Asia
Indic:Southeast Asia
Europe:Inner Asia
Greater Mesopotamia:N Coast Asia
N Coast Asia:N Coast Asia
Europe:Greater Mesopotamia
Greater Mesopotamia:Indic
Europe:Indic
Indic:N Coast Asia
Indic:Indic
Greater Mesopotamia:Greater Mesopotamia
Greater Mesopotamia:Inner Asia
Inner Asia:Inner Asia
Indic:Inner Asia
−1
−0.5
0
0.5
1
−1
−0.5
0
0.5
1

Without Kinship data

The analyses below show that the strongest relationship is with Kinship. Here we run the analysis as above, but using semantic distances computed without concepts that relate to kinship. Note that the local alignment values correlate with r > 0.99.

Code for constructing the data is hidden, but it is the same as above and available in the Rmd file:

Run the lmer models:

```
m0NK = lmer(
  rho.center ~ 1 +
    (1 + cult.dist.center | family.group) +
    (1 + cult.dist.center | area.group),
  data = lingNK
)
```

```
## boundary (singular) fit: see ?isSingular
```

```
m0.5NK = lmer(
  rho.center ~ 1 +
    comparison_count.center +
    (1 + cult.dist.center | family.group) +
    (1 + cult.dist.center | area.group),
  data = lingNK
)
```

```
## boundary (singular) fit: see ?isSingular
```

```
m1NK = lmer(
  rho.center ~ 1 +
    comparison_count.center +
    cult.dist.center +
    (1 + cult.dist.center | family.group) +
    (1 + cult.dist.center | area.group),
  data = lingNK
)
```

```
## boundary (singular) fit: see ?isSingular
```

```
anova(m0NK,m0.5NK,m1NK)
```

```
## refitting model(s) with ML (instead of REML)
```

```
## Data: lingNK
## Models:
## m0NK: rho.center ~ 1 + (1 + cult.dist.center | family.group) + (1 +
## m0NK:     cult.dist.center | area.group)
## m0.5NK: rho.center ~ 1 + comparison_count.center + (1 + cult.dist.center |
## m0.5NK:     family.group) + (1 + cult.dist.center | area.group)
## m1NK: rho.center ~ 1 + comparison_count.center + cult.dist.center +
## m1NK:     (1 + cult.dist.center | family.group) + (1 + cult.dist.center |
## m1NK:     area.group)
##        Df    AIC    BIC  logLik deviance   Chisq Chi Df Pr(>Chisq)
## m0NK    8 1654.6 1691.3 -819.30   1638.6
## m0.5NK  9 1293.0 1334.3 -637.50   1275.0 363.597      1  < 2.2e-16 ***
## m1NK   10 1278.4 1324.4 -629.22   1258.4  16.564      1  4.704e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(m1NK)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: rho.center ~ 1 + comparison_count.center + cult.dist.center +
##     (1 + cult.dist.center | family.group) + (1 + cult.dist.center |
##     area.group)
##    Data: lingNK
##
## REML criterion at convergence: 1271.9
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.6249 -0.6167  0.1103  0.6571  4.7402
##
## Random effects:
##  Groups       Name             Variance  Std.Dev. Corr
##  family.group (Intercept)      0.1612630 0.40158
##               cult.dist.center 0.0001817 0.01348  1.00
##  area.group   (Intercept)      0.0510850 0.22602
##               cult.dist.center 0.0036658 0.06055  -1.00
##  Residual                      0.2885416 0.53716
## Number of obs: 731, groups:  family.group, 48; area.group, 20
##
## Fixed effects:
##                          Estimate Std. Error t value
## (Intercept)             -0.39270    0.09073  -4.328
## comparison_count.center  0.61196    0.02688  22.770
## cult.dist.center         0.19678    0.03275   6.008
##
## Correlation of Fixed Effects:
##             (Intr) cmpr_.
## cmprsn_cnt.  0.090
## clt.dst.cnt -0.194 -0.201
## convergence code: 0
## boundary (singular) fit: see ?isSingular
```

MRM

Use multiple regression on distance matrices (Lichstein, 2007) to do the same test as above. The code below uses the `igraph` package to make an undirected graph from the long format with `local_alignment` as the edge weights, then output a matrix of adjacencies.

```
# Use graph method to make distance matrix
grph <- graph.data.frame(ling[,c("l1",'l2','local_alignment')], directed=FALSE)
# add value as a weight attribute
ling.m = get.adjacency(grph, attr="local_alignment", sparse=FALSE)
rownames(ling.m) = l[match(rownames(ling.m),l$iso2),]$Language2
colnames(ling.m) = l[match(colnames(ling.m),l$iso2),]$Language2
# Same for comparison_count.center
grph <- graph.data.frame(ling[,c("l1",'l2','comparison_count')], directed=FALSE)
# add value as a weight attribute
cc.m = get.adjacency(grph, attr="comparison_count", sparse=FALSE)
rownames(cc.m) = l[match(rownames(cc.m),l$iso2),]$Language2
colnames(cc.m) = l[match(colnames(cc.m),l$iso2),]$Language2
```

Load the cultural distances as a matrix.

```
cult.m = read.csv("../results/EA_distances/CulturalDistances.csv", stringsAsFactors = F)
rownames(cult.m) = cult.m[,1]
cult.m = cult.m[,2:ncol(cult.m)]
cult.m = as.matrix(cult.m)
# Flip cultural value to distance
cult.m = 1-cult.m
mx = match(rownames(ling.m),rownames(cult.m))
cult.m = cult.m[mx,mx]
colnames(cult.m) = rownames(cult.m)
```

Make a matrix of same/different language family (1=different):

```
# Same/different matrix for language family
family.matrix = l[match(rownames(ling.m),l$Language),]$family
family.matrix = outer(family.matrix,family.matrix,"!=") *1
```

Load ASJP distances for second test:

```
asjp = readRDS("../data/ASJP/asjp17-dists_FAIR.RData")
ling.m.glotto = l[match(rownames(cult.m),l$Language2),]$glotto
ling.m.glotto = ling.m.glotto[ling.m.glotto %in% rownames(asjp)]
asjp.m = asjp[ling.m.glotto,ling.m.glotto]
asjp.lang.names = l[match(rownames(asjp.m),l$glotto),]$Language2
# Matrices for second analysis with asjp
ling.m2 = ling.m[asjp.lang.names,asjp.lang.names]
cult.m2 = cult.m[asjp.lang.names,asjp.lang.names]
cc.m2 =  cc.m[asjp.lang.names,asjp.lang.names]
```

Load the geographic distances:

```
geoDist = read.csv("../data/GeographicDistances.csv",stringsAsFactors = F)
geoDist.m = as.matrix(geoDist)
geoDist.m = geoDist.m[!is.na(geoDist.m[,1]),!is.na(geoDist.m[1,])]
# Convert to log distance in thousand km
geoDist.m = log10(geoDist.m/1000)
geoDist.m[is.infinite(geoDist.m)] = 0
```

```
colnames(geoDist.m) = gsub("\\."," ",colnames(geoDist.m))
rownames(geoDist.m) = colnames(geoDist.m)
geoDist.m1 = geoDist.m[rownames(ling.m),rownames(ling.m)]
geoDist.m2 = geoDist.m[rownames(ling.m2),rownames(ling.m2)]
```

Some language pairs do not have observed semantic alignments (10 out of 741, 1.3%). In this case, we impute the mean:

```
# For missing comparisons, impute the mean:
# (there are no zero values in the local alignment data)
ling.m[ling.m==0] = mean(ling$local_alignment)
diag(ling.m) = 0
ling.m2[ling.m2==0] = mean(ling.m2[ling.m2!=0])
diag(ling.m2) = 0
```

Center and scale values:

```
ling.m =  matrix(scale(as.vector(ling.m)),nrow=nrow(ling.m))
cc.m =  matrix(scale(as.vector(cc.m)),nrow=nrow(cc.m))
cult.m = matrix(scale(as.vector(cult.m)),nrow=nrow(cult.m))
geoDist.m1 = matrix(scale(as.vector(geoDist.m1)),nrow=nrow(geoDist.m1))

asjp.m = matrix(scale(as.vector(asjp.m)),nrow=nrow(asjp.m))
ling.m2 = matrix(scale(as.vector(ling.m2)),nrow=nrow(ling.m2))
cc.m2 =  matrix(scale(as.vector(cc.m2)),nrow=nrow(cc.m2))
cult.m2 = matrix(scale(as.vector(cult.m2)),nrow=nrow(cult.m2))
geoDist.m2 = matrix(scale(as.vector(geoDist.m2)),nrow=nrow(geoDist.m2))
```

Run the MRM model, predicting semantic alignment by cultural distance, controlling for family distance, geographic ditance, and the comparison count (number of observations). Here, the family distance between two languages is just whether they are part of the same family. Note that this does not take into account particular values for particular families, nor the random slopes within families.

```
set.seed(1282)
ecodist::MRM(as.dist(ling.m) ~
               as.dist(cult.m) +
               as.dist(family.matrix) +
               as.dist(geoDist.m1)  +
               as.dist(cc.m),nperm = 10000)
```

```
## $coef
##                          as.dist(ling.m)   pval
## Int                           0.22588260 0.0512
## as.dist(cult.m)               0.27484467 0.0114
## as.dist(family.matrix)       -0.21726450 0.1267
## as.dist(geoDist.m1)          -0.05699071 0.4481
## as.dist(cc.m)                 0.56172533 0.0001
##
## $r.squared
##        R2      pval
## 0.5698601 0.0001000
##
## $F.test
##        F   F.pval
## 243.7678   0.0001
```

Semantic alignment is significantly correlated with cultural distance.

In the result above, geographic distance is not correlated with semantic distance. Geographic distance turns out to be moderately correlated with cultural distance:

```
ecodist::MRM(as.dist(geoDist.m1) ~
               as.dist(cult.m),
             nperm = 10000)
```

```
## $coef
##                   as.dist(geoDist.m1)   pval
## Int                        -0.02111026 0.7707
## as.dist(cult.m)            -0.50170516 0.0001
##
## $r.squared
##        R2      pval
## 0.1529605 0.0001000
##
## $F.test
##        F   F.pval
## 133.4504   0.0001
```

Even when testing for non-linear geographic effects, the main result still holds:

```
ecodist::MRM(as.dist(ling.m) ~
               as.dist(cult.m) +
               as.dist(family.matrix) +
               as.dist(geoDist.m1)  +
               as.dist(geoDist.m1^2)  +
```

```
              as.dist(geoDist.m1^3)  +
              as.dist(cc.m),nperm = 10000)
```

```
## $coef
##                         as.dist(ling.m)   pval
## Int                          0.208052336 0.0865
## as.dist(cult.m)              0.272562935 0.0134
## as.dist(family.matrix)      -0.229500368 0.1162
## as.dist(geoDist.m1)         -0.018785241 0.8384
## as.dist(geoDist.m1^2)        0.021357131 0.7142
## as.dist(geoDist.m1^3)       -0.006518993 0.7250
## as.dist(cc.m)                0.563554206 0.0001
##
## $r.squared
##        R2      pval
## 0.5730241 0.0001000
##
## $F.test
##        F   F.pval
## 164.1777   0.0001
```

Below, we run the same test, but using average string distances in basic vocabulary from the ASJP (Wichmann, Holman & Brown, 2018) as controls for history. We used the distances as calculated in Jäger (2018), which used them to construct historical phylogenies.

```
ecodist::MRM(as.dist(ling.m2) ~
              as.dist(cult.m2) +
              as.dist(asjp.m) +
              as.dist(geoDist.m2) +
              as.dist(cc.m2),nperm = 10000)
```

```
## $coef
##                       as.dist(ling.m2)   pval
## Int                        0.10125375 0.0005
## as.dist(cult.m2)           0.26218752 0.0235
## as.dist(asjp.m)           -0.24422112 0.0001
## as.dist(geoDist.m2)       -0.02893256 0.7035
## as.dist(cc.m2)             0.56824024 0.0001
##
## $r.squared
##        R2      pval
## 0.5782528 0.0001000
##
## $F.test
##        F   F.pval
## 214.2326   0.0001
```

Mantel tests

Read the historical distances for Indo-European, based on the phylogenetic distances.

Data prep

The geographic distances are loaded above (from "../data/GeographicDistances.csv").

Load historical distances (Indo-European tree patristic distances):

```
hist = read.csv("../data/trees/IndoEuropean_historical_distances.csv", stringsAsFactors = F)
hist = hist[!duplicated(hist[,1]),!duplicated(hist[,1])]
rownames(hist) = hist[,1]
hist = hist[,2:ncol(hist)]
hist.m = as.matrix(hist)
colnames(hist.m) = rownames(hist.m)
hist.m = hist.m/max(hist.m)
```

Read the cultural distance as a matrix:

```
cult.m = read.csv("../results/EA_distances/CulturalDistances.csv", stringsAsFactors = F)
rownames(cult.m) = cult.m[,1]
cult.m = cult.m[,2:ncol(cult.m)]
```

Flip the cultural distance into a cultural similarity measure:

```
cult.m = 1-cult.m
```

Convert the semantic alignment to a matrix and impute the missing values with the mean. Note that in the final selection of langauges excludes any imputed values, but we perform the imputation just to be safe:

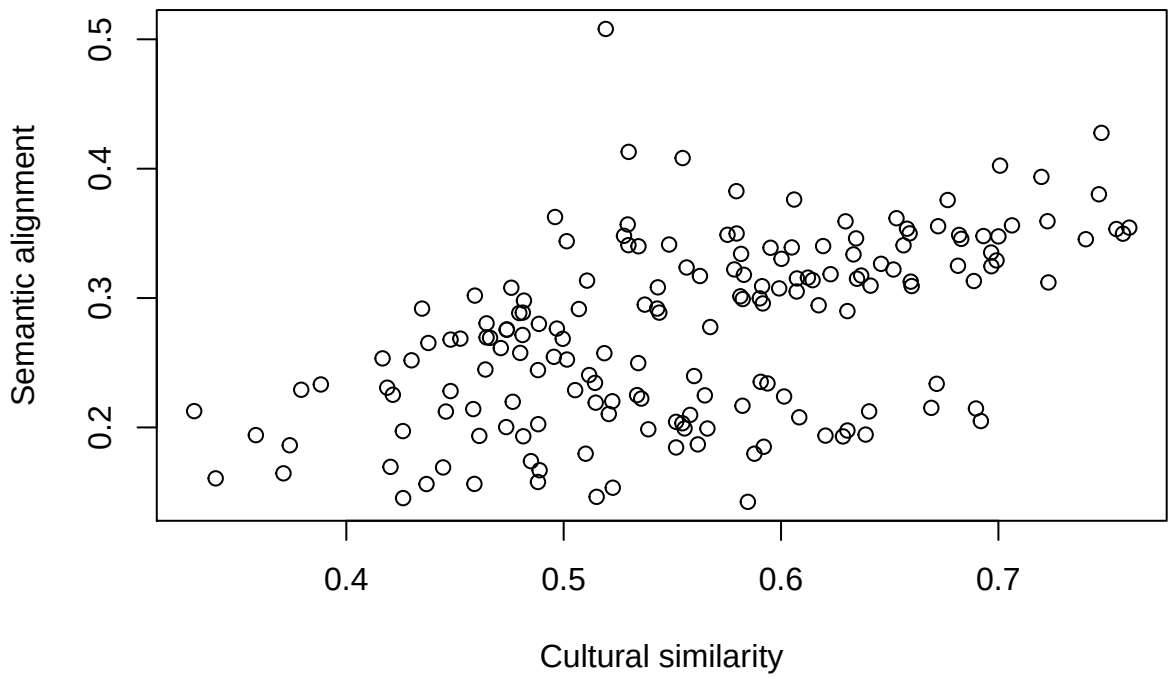
```
grph <- graph.data.frame(ling[,c("l1",'l2','local_alignment')], directed=FALSE)
# add value as a weight attribute
ling.m = get.adjacency(grph, attr="local_alignment", sparse=FALSE)
rownames(ling.m) = l[match(rownames(ling.m),l$iso2),]$Language2
colnames(ling.m) = l[match(colnames(ling.m),l$iso2),]$Language2
# For missing comparisons, impute the mean:
# (there are no zero values in the local alignment data)
ling.m[ling.m==0] = mean(ling$local_alignment)
diag(ling.m) = 0
```

Match the distance matrices

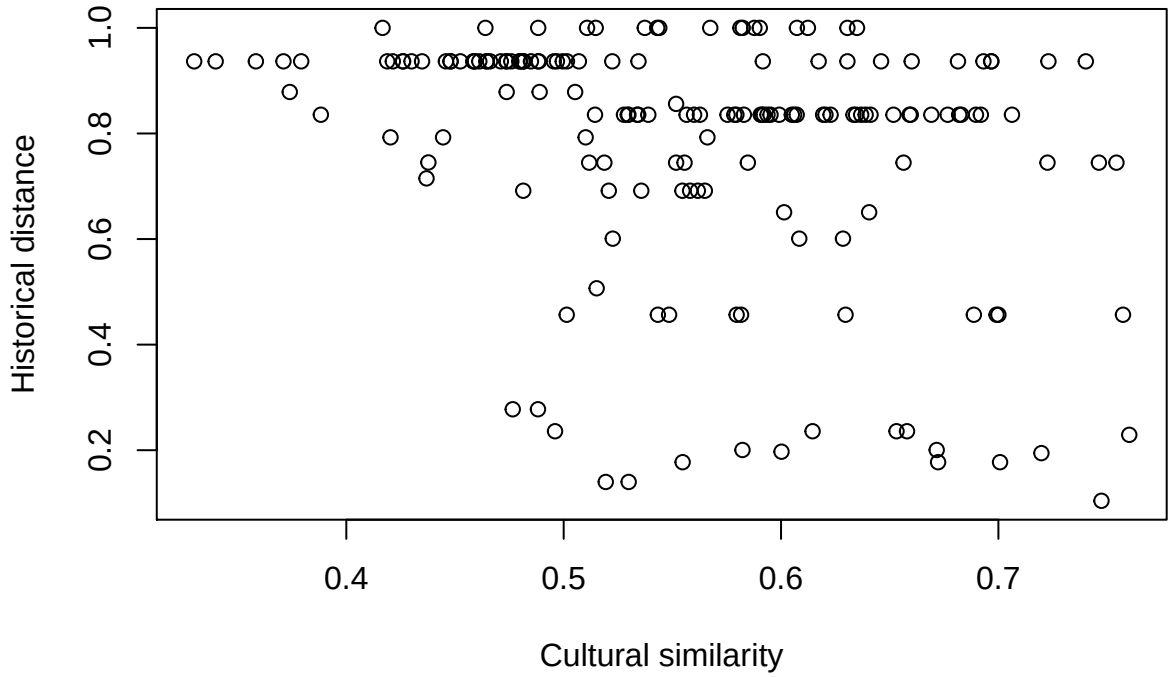
```
in.analysis = intersect(rownames(ling.m),rownames(cult.m))
in.analysis = intersect(in.analysis, rownames(hist.m))
cult.m2 = cult.m[in.analysis,in.analysis]
ling.m2 = ling.m[in.analysis,in.analysis]
hist.m2 = hist.m[in.analysis,in.analysis]
geo.m2 = geoDist.m[in.analysis,in.analysis]
```

Note that there are only 19 languages with data on lingusitic, cultural and historical distance. This is becasue the historical distances are derived from a tree of Indo-European languages (there are currently no reliable phylogentic trees constructed from cognates that span different language families). The languages in this test include: Albanian, Armenian, Belarusian, Bengali, Bulgarian, Czech, Dutch, English, French, Greek, Icelandic, Irish, Latin, Latvian, Lithuanian, Ossetian, Russian, Spanish, Ukrainian.

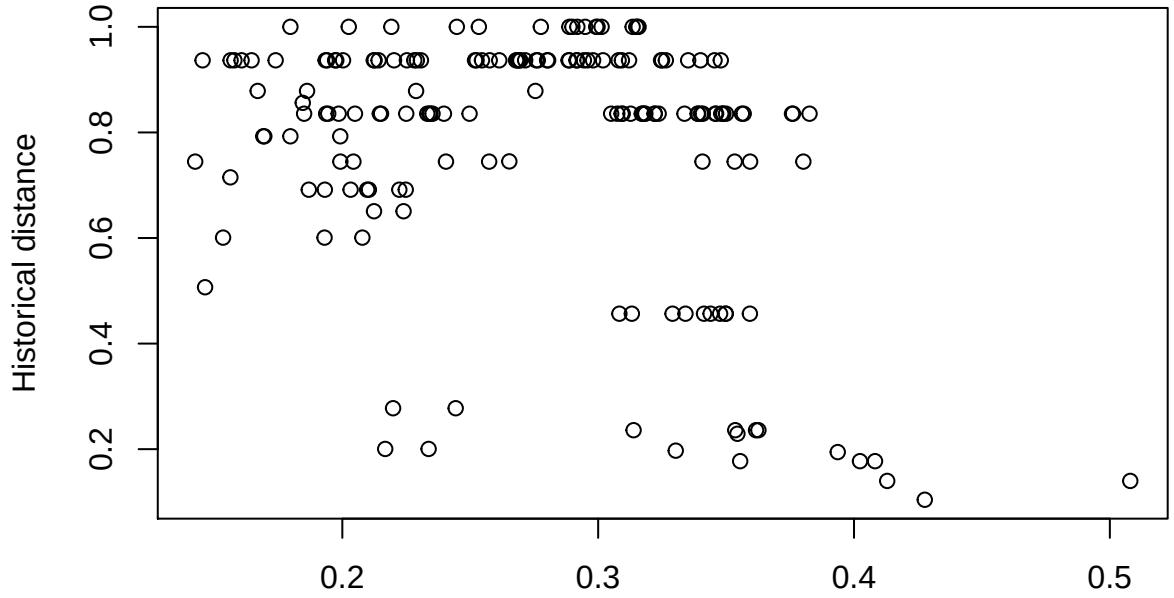
```
plot(as.dist(cult.m2),as.dist(ling.m2),
     xlab="Cultural similarity",
     ylab="Semantic alignment")
```

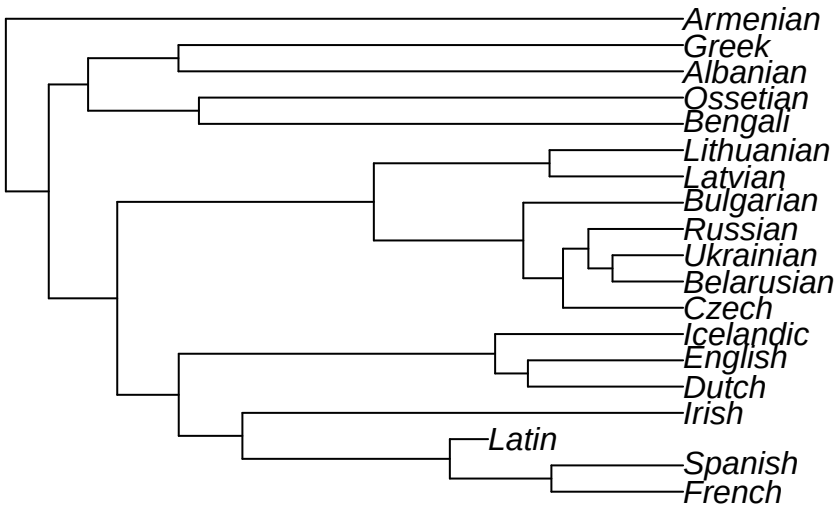



```
plot(as.dist(cult.m2),as.dist(hist.m2),
     xlab="Cultural similarity",
     ylab="Historical distance")
```




```
plot(as.dist(ling.m2),as.dist(hist.m2),
     xlab="Semantic alignment",
     ylab="Historical distance")
```


Historical distance
0.2
0.4
0.6
0.8
1.0
0.2
0.3
0.4
0.5
Semantic alignment


Armenian
Greek
Albanian
Ossetian
Bengali
Lithuanian
Latvian
Bulgarian
Russian
Ukrainian
Belarusian
Czech
Icelandic
English
Dutch
Irish
Latin
Spanish
French

Tests

The results of the test list the following measures:

- mantelr: Mantel correlation coefficient.
- pval1: one-tailed p-value (null hypothesis: r <= 0).
- pval2: one-tailed p-value (null hypothesis: r >= 0).
- pval3: two-tailed p-value (null hypothesis: r = 0).
- llim: lower confidence limit for r.
- ulim: upper confidence limit for r.

```
set.seed(1498)
```

Run tests between each pair of measures.

```
distms = list("Cultrual"= cult.m2,
              "Linguistic" = ling.m2,
              "Historical" = hist.m2,
              "Geographic" = geo.m2)
for(i in 1:3){
  for(j in (i+1):4){
    var1 = names(distms)[i]
    var2 = names(distms)[j]
    print(paste("Correlation between",
                var1,"and",var2))
    stat = ecodist::mantel(as.dist(distms[[i]]) ~
                as.dist(distms[[j]]),
                nperm = 100000)
    print(stat)
    stat = round(stat,2)
    pval = round(min(c(stat[2],stat[3])),3)
    if(pval==0){pval = "$<$ 0.001"}
    stat2 = sprintf("$r$ = %s, 95\\%% CI = [%s,%s], one-tailed $p$ = %s",
      round(stat[1],3),
      round(stat[5],3),
      round(stat[6],3),
      pval)
    stat2 = gsub("0\\.",".",stat2)
    cat(stat2,file=
          paste0("../results/stats/tex/Mantel",var1,"Vs",var2,"Distance.tex"))
  }
}
```

```
## [1] "Correlation between Cultrual and Linguistic"
##    mantelr      pval1      pval2      pval3  llim.2.5% ulim.97.5%
##  0.5243289  0.0050000  0.9950100  0.0050300  0.3796035  0.6586819
## [1] "Correlation between Cultrual and Historical"
##    mantelr      pval1      pval2      pval3  llim.2.5% ulim.97.5%
## -0.3243830  0.9871000  0.0129100  0.0138900 -0.4402666 -0.2385575
## [1] "Correlation between Cultrual and Geographic"
##    mantelr      pval1      pval2      pval3  llim.2.5% ulim.97.5%
## -0.4495398  0.9967200  0.0032900  0.0032900 -0.5754918 -0.3109193
## [1] "Correlation between Linguistic and Historical"
##    mantelr      pval1      pval2      pval3  llim.2.5% ulim.97.5%
## -0.3372882  0.9859600  0.0140500  0.0167300 -0.5019408 -0.1639425
## [1] "Correlation between Linguistic and Geographic"
```

```
##    mantelr      pval1      pval2      pval3  llim.2.5% ulim.97.5%
## -0.2594386  0.9182700  0.0817400  0.1195200 -0.3694719 -0.1840035
## [1] "Correlation between Historical and Geographic"
##    mantelr      pval1      pval2      pval3  llim.2.5% ulim.97.5%
##  0.4210629  0.0004100  0.9996000  0.0004100  0.3313578  0.5176683
```

Run a mantel test comparing the semantic alignment to the cultural similarity, controlling for the historical distance between languages:

```
ecodist::mantel(as.dist(ling.m2)~
                  as.dist(cult.m2) +
                  as.dist(hist.m2),
                nperm = 100000)
```

```
##    mantelr      pval1      pval2      pval3  llim.2.5% ulim.97.5%
##  0.4659407  0.0100000  0.9900100  0.0107800  0.3408500  0.5938397
```

Main Test: Run a mantel test comparing the semantic alignment to the cultural similarity, controlling for the historical distance and geographic distance between languages:

```
mainMantel = ecodist::mantel(as.dist(ling.m2)~
                  as.dist(cult.m2) +
                  as.dist(hist.m2) +
                  as.dist(geo.m2),
                nperm = 100000)
mainMantel
```

```
##    mantelr      pval1      pval2      pval3  llim.2.5% ulim.97.5%
##  0.4508309  0.0114200  0.9885900  0.0119500  0.2962271  0.5993660
```

```
mainMantel = round(mainMantel,2)
mainMantel2 = sprintf("$r$ = %s, 95\\%% CI = [%s,%s], one-tailed $p$ = %s",
        round(mainMantel[1],3),
        round(mainMantel[5],3),
        round(mainMantel[6],3),
        round(mainMantel[2],3)
      )
mainMantel2 = gsub("0\\.",".",mainMantel2)
cat(mainMantel2,
    file="../results/stats/tex/MantelCultrualVsLinguisticDistance_Partial.tex")
```

MRM

Perform the main test using the phylogenetic distance, but using multiple regression on distance matrices (MRM).

```
set.seed(21889)
mainMRM = ecodist::MRM(as.dist(ling.m2)~
                  as.dist(cult.m2) +
                  as.dist(hist.m2) +
                  as.dist(geo.m2), nperm=10000)
mainMRM
```

```
## $coef
##                   as.dist(ling.m2)   pval
## Int                    0.123122021 0.9147
## as.dist(cult.m2)       0.350192033 0.0108
```

```
## as.dist(hist.m2)     -0.059044640 0.1623
## as.dist(geo.m2)       0.008177519 0.8212
##
## $r.squared
##        R2      pval
## 0.3072419 0.0073000
##
## $F.test
##        F   F.pval
## 24.68846  0.00730
```

```
mainMRM2 = sprintf("$\\beta= $%s, $p=$%s",
                   round(mainMRM$coef[2,1],2),
                   round(mainMRM$coef[2,2],2))
cat(mainMRM2,
    file="../results/stats/tex/MRMCultrualVsLinguisticDistance_Partial.tex")
```

Analysis of filtered data

The analyses in this section use local alignment values based on (a) data that passes the wikipedia filter, and (b) data that passes the semantic filter.

Wikipedia filter

```
ling.filtered = read.csv(
  "../data/FAIR/nel-wiki-k100-alignments-by-language-pair_Filtered.csv",
  stringsAsFactors = F)
```

Note that the semantic alignment for the filtered and unfiltered data are essentially exactly the same, but for fewer languages:

```
ling.filtered$unfiltered.rho =
  apply(ling.filtered[,
      c("iso2_l1","iso2_l2")],1,
  function(X){
    ling[(ling$l1==X[1] & ling$l2==X[2]) |
           (ling$l1==X[2] & ling$l2==X[1]),]$local_alignment[1]
  })
cor(ling.filtered$unfiltered.rho,ling.filtered$rho,use = "complete.obs")
```

```
## [1] 0.9999918
```

Continue to build data for replication:

```
ling.filtered$area1 = l[match(ling.filtered$name_l1,l$Language),]$autotyp.area
ling.filtered$area2 = l[match(ling.filtered$name_l2,l$Language),]$autotyp.area

fgroup = cbind(ling.filtered$family1,ling.filtered$family2)
fgroup = apply(fgroup,1,sort)
ling.filtered$family.group = apply(fgroup,2,paste,collapse=":")
agroup = cbind(ling.filtered$area1,ling.filtered$area2)
agroup = apply(agroup,1,sort)
ling.filtered$area.group = apply(agroup,2,paste,collapse=":")

ling.filtered$rho.center = scale(ling.filtered$rho)
ling.filtered$comparison_count.center = scale(ling.filtered$comparison_count)

matches = sapply(1:nrow(ling.filtered), function(i){
  x = which((cult$l1==ling.filtered$name_l1[i] &
          cult$l2==ling.filtered$name_l2[i]) |
            (cult$l2==ling.filtered$name_l1[i] &
          cult$l1==ling.filtered$name_l2[i]))
  x[1]
})

ling.filtered$cult.dist = cult[matches,]$cult.dist
# flip
ling.filtered$cult.dist = 1 - ling.filtered$cult.dist
ling.filtered = ling.filtered[!is.na(ling.filtered$cult.dist),]

ling.filtered$cult.dist.center = scale(ling.filtered$cult.dist)
```

```
m0F = lmer(
  rho.center ~ 1 +
    (1 + cult.dist.center | family.group) +
    (1 + cult.dist.center | area.group),
  data = ling.filtered
)
```

```
## boundary (singular) fit: see ?isSingular
```

```
m0.5F = lmer(
  rho.center ~ 1 +
    comparison_count.center +
    (1 + cult.dist.center | family.group) +
    (1 + cult.dist.center | area.group),
  data = ling.filtered
)
```

```
## boundary (singular) fit: see ?isSingular
```

```
m1F = lmer(
  rho.center ~ 1 +
    comparison_count.center +
    cult.dist.center +
    (1 + cult.dist.center | family.group) +
    (1 + cult.dist.center | area.group),
  data = ling.filtered
)
an1F = anova(m0F,m0.5F,m1F)
```

```
## refitting model(s) with ML (instead of REML)
```

```
an1F
```

```
## Data: ling.filtered
## Models:
## m0F: rho.center ~ 1 + (1 + cult.dist.center | family.group) + (1 +
## m0F:     cult.dist.center | area.group)
## m0.5F: rho.center ~ 1 + comparison_count.center + (1 + cult.dist.center |
## m0.5F:     family.group) + (1 + cult.dist.center | area.group)
## m1F: rho.center ~ 1 + comparison_count.center + cult.dist.center +
## m1F:     (1 + cult.dist.center | family.group) + (1 + cult.dist.center |
## m1F:     area.group)
##       Df    AIC    BIC  logLik deviance   Chisq Chi Df Pr(>Chisq)
## m0F    8 446.96 473.74 -215.48   430.96
## m0.5F  9 428.63 458.76 -205.32   410.63 20.3271      1  6.527e-06 ***
## m1F   10 425.68 459.15 -202.84   405.68  4.9519      1    0.02606 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Cultural similarity is significantly correlated with semantic alignment, even in the filtered data. Here are the model estimates:

```
summary(m1F)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: rho.center ~ 1 + comparison_count.center + cult.dist.center +
##     (1 + cult.dist.center | family.group) + (1 + cult.dist.center |
##     area.group)
```

```
##    Data: ling.filtered
##
## REML criterion at convergence: 412.8
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.8211 -0.5105 -0.0662  0.4157  6.2698
##
## Random effects:
##  Groups       Name             Variance Std.Dev. Corr
##  family.group (Intercept)      0.17449  0.4177
##               cult.dist.center 0.07149  0.2674   -0.21
##  area.group   (Intercept)      0.64616  0.8038
##               cult.dist.center 0.02791  0.1670   1.00
##  Residual                      0.26481  0.5146
## Number of obs: 210, groups:  family.group, 31; area.group, 19
##
## Fixed effects:
##                          Estimate Std. Error t value
## (Intercept)             -0.82996    0.21638  -3.836
## comparison_count.center  0.41070    0.08668   4.738
## cult.dist.center         0.29681    0.11189   2.653
##
## Correlation of Fixed Effects:
##             (Intr) cmpr_.
## cmprsn_cnt.  0.133
## clt.dst.cnt  0.387 -0.054
```

Semantic filter

```
ling.semFiltered = read.csv(
  "../data/FAIR/nel-wiki-k100-alignments-by-language-pair_SemanticFiltered.csv",
  stringsAsFactors = F)

ling.semFiltered$area1 = l[match(ling.semFiltered$name_l1,l$Language),]$autotyp.area
ling.semFiltered$area2 = l[match(ling.semFiltered$name_l2,l$Language),]$autotyp.area

fgroup = cbind(ling.semFiltered$family1,ling.semFiltered$family2)
fgroup = apply(fgroup,1,sort)
ling.semFiltered$family.group = apply(fgroup,2,paste,collapse=":")
agroup = cbind(ling.semFiltered$area1,ling.semFiltered$area2)
agroup = apply(agroup,1,sort)
ling.semFiltered$area.group = apply(agroup,2,paste,collapse=":")

ling.semFiltered$rho.center = scale(ling.semFiltered$rho)
ling.semFiltered$comparison_count.center = scale(ling.semFiltered$comparison_count)

matches = sapply(1:nrow(ling.semFiltered), function(i){
  x = which((cult$l1==ling.semFiltered$name_l1[i] &
          cult$l2==ling.semFiltered$name_l2[i]) |
            (cult$l2==ling.semFiltered$name_l1[i] &
          cult$l1==ling.semFiltered$name_l2[i]))
  x[1]
})

ling.semFiltered$cult.dist = cult[matches,]$cult.dist
# flip
ling.semFiltered$cult.dist = 1 - ling.semFiltered$cult.dist
ling.semFiltered = ling.semFiltered[!is.na(ling.semFiltered$cult.dist),]

ling.semFiltered$cult.dist.center = scale(ling.semFiltered$cult.dist)
```

```
m0F = lmer(
  rho.center ~ 1 +
    (1 + cult.dist.center | family.group) +
    (1 + cult.dist.center | area.group),
  data = ling.semFiltered
)
```

```
## boundary (singular) fit: see ?isSingular
```

```
m0.5F = lmer(
  rho.center ~ 1 +
    comparison_count.center +
    (1 + cult.dist.center | family.group) +
    (1 + cult.dist.center | area.group),
  data = ling.semFiltered
)
```

```
## boundary (singular) fit: see ?isSingular
```

```
m1F = lmer(
  rho.center ~ 1 +
```

```
    comparison_count.center +
    cult.dist.center +
    (1 + cult.dist.center | family.group) +
    (1 + cult.dist.center | area.group),
  data = ling.semFiltered
)
```

```
## boundary (singular) fit: see ?isSingular
```

```
an1F = anova(m0F,m0.5F,m1F)
```

```
## refitting model(s) with ML (instead of REML)
```

```
an1F
```

```
## Data: ling.semFiltered
## Models:
## m0F: rho.center ~ 1 + (1 + cult.dist.center | family.group) + (1 +
## m0F:     cult.dist.center | area.group)
## m0.5F: rho.center ~ 1 + comparison_count.center + (1 + cult.dist.center |
## m0.5F:     family.group) + (1 + cult.dist.center | area.group)
## m1F: rho.center ~ 1 + comparison_count.center + cult.dist.center +
## m1F:     (1 + cult.dist.center | family.group) + (1 + cult.dist.center |
## m1F:     area.group)
##       Df    AIC    BIC  logLik deviance   Chisq Chi Df Pr(>Chisq)
## m0F    8 1660.3 1697.1 -822.17   1644.3
## m0.5F  9 1300.8 1342.1 -641.39   1282.8 361.576      1  < 2.2e-16 ***
## m1F   10 1286.2 1332.1 -633.09   1266.2  16.595      1  4.627e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Cultural similarity is significantly correlated with semantic alignment, even in the semantic filtered data. Here are the model estimates:

```
summary(m1F)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: rho.center ~ 1 + comparison_count.center + cult.dist.center +
##     (1 + cult.dist.center | family.group) + (1 + cult.dist.center |
##     area.group)
##    Data: ling.semFiltered
##
## REML criterion at convergence: 1279.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.6257 -0.6155  0.1002  0.6528  4.7581
##
## Random effects:
##  Groups       Name             Variance  Std.Dev. Corr
##  family.group (Intercept)      0.1627947 0.40348
##               cult.dist.center 0.0001944 0.01394  1.00
##  area.group   (Intercept)      0.0515199 0.22698
##               cult.dist.center 0.0037519 0.06125  -1.00
##  Residual                      0.2916401 0.54004
## Number of obs: 731, groups:  family.group, 48; area.group, 20
##
```

```
## Fixed effects:
##                           Estimate Std. Error t value
## (Intercept)              -0.39039    0.09111  -4.285
## comparison_count.center   0.62768    0.02766  22.696
## cult.dist.center          0.19836    0.03301   6.009
##
## Correlation of Fixed Effects:
##             (Intr) cmpr_.
## cmprsn_cnt.  0.083
## clt.dst.cnt -0.192 -0.200
## convergence code: 0
## boundary (singular) fit: see ?isSingular
```

Both filters

Main test on data where both the wikipedia and semantic filter are on.

```
ling.bothFiltered = read.csv(
  "../data/FAIR/nel-wiki-k100-alignments-by-language-pair_BothFiltered.csv",
  stringsAsFactors = F)

ling.bothFiltered$area1 = l[match(ling.bothFiltered$name_l1,l$Language),]$autotyp.area
ling.bothFiltered$area2 = l[match(ling.bothFiltered$name_l2,l$Language),]$autotyp.area

fgroup = cbind(ling.bothFiltered$family1,ling.bothFiltered$family2)
fgroup = apply(fgroup,1,sort)
ling.bothFiltered$family.group = apply(fgroup,2,paste,collapse=":")
agroup = cbind(ling.bothFiltered$area1,ling.bothFiltered$area2)
agroup = apply(agroup,1,sort)
ling.bothFiltered$area.group = apply(agroup,2,paste,collapse=":")

ling.bothFiltered$rho.center = scale(ling.bothFiltered$rho)
ling.bothFiltered$comparison_count.center = scale(ling.bothFiltered$comparison_count)

matches = sapply(1:nrow(ling.bothFiltered), function(i){
  x = which((cult$l1==ling.bothFiltered$name_l1[i] &
          cult$l2==ling.bothFiltered$name_l2[i]) |
            (cult$l2==ling.bothFiltered$name_l1[i] &
          cult$l1==ling.bothFiltered$name_l2[i]))
  x[1]
})

ling.bothFiltered$cult.dist = cult[matches,]$cult.dist
# flip
ling.bothFiltered$cult.dist = 1 - ling.bothFiltered$cult.dist
ling.bothFiltered = ling.bothFiltered[!is.na(ling.bothFiltered$cult.dist),]

ling.bothFiltered$cult.dist.center = scale(ling.bothFiltered$cult.dist)
```

```
m0F = lmer(
  rho.center ~ 1 +
    (1 + cult.dist.center | family.group) +
    (1 + cult.dist.center | area.group),
  data = ling.bothFiltered
)
```

```
## boundary (singular) fit: see ?isSingular
```

```
m0.5F = lmer(
  rho.center ~ 1 +
    comparison_count.center +
    (1 + cult.dist.center | family.group) +
    (1 + cult.dist.center | area.group),
  data = ling.bothFiltered
)
```

```
## boundary (singular) fit: see ?isSingular
```

```
m1F = lmer(
  rho.center ~ 1 +
    comparison_count.center +
    cult.dist.center +
    (1 + cult.dist.center | family.group) +
    (1 + cult.dist.center | area.group),
  data = ling.bothFiltered
)
an1F = anova(m0F,m0.5F,m1F)
```

```
## refitting model(s) with ML (instead of REML)
```

```
an1F
```

```
## Data: ling.bothFiltered
## Models:
## m0F: rho.center ~ 1 + (1 + cult.dist.center | family.group) + (1 +
## m0F:     cult.dist.center | area.group)
## m0.5F: rho.center ~ 1 + comparison_count.center + (1 + cult.dist.center |
## m0.5F:     family.group) + (1 + cult.dist.center | area.group)
## m1F: rho.center ~ 1 + comparison_count.center + cult.dist.center +
## m1F:     (1 + cult.dist.center | family.group) + (1 + cult.dist.center |
## m1F:     area.group)
##       Df    AIC    BIC  logLik deviance   Chisq Chi Df Pr(>Chisq)
## m0F    8 447.24 474.02 -215.62   431.24
## m0.5F  9 429.45 459.58 -205.73   411.45 19.7903      1  8.642e-06 ***
## m1F   10 426.47 459.95 -203.24   406.47  4.9762      1     0.0257 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Cultural similarity is significantly correlated with semantic alignment, even in the fully filtered data. Here are the model estimates:

```
summary(m1F)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: rho.center ~ 1 + comparison_count.center + cult.dist.center +
##     (1 + cult.dist.center | family.group) + (1 + cult.dist.center |
##     area.group)
##    Data: ling.bothFiltered
##
## REML criterion at convergence: 413.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.8275 -0.5144 -0.0659  0.4117  6.2874
##
## Random effects:
##  Groups       Name             Variance Std.Dev. Corr
##  family.group (Intercept)      0.17288  0.4158
##               cult.dist.center 0.07138  0.2672   -0.21
##  area.group   (Intercept)      0.64816  0.8051
##               cult.dist.center 0.02802  0.1674   1.00
##  Residual                      0.26612  0.5159
## Number of obs: 210, groups:  family.group, 31; area.group, 19
##
```

```
## Fixed effects:
##                           Estimate Std. Error t value
## (Intercept)               -0.8297     0.2166  -3.831
## comparison_count.center    0.4075     0.0872   4.673
## cult.dist.center           0.2976     0.1119   2.659
##
## Correlation of Fixed Effects:
##             (Intr) cmpr_.
## cmprsn_cnt.  0.135
## clt.dst.cnt  0.387 -0.053
```

Comparison between domains

The code that produce the results of this section can be found in `analysis/compareDomains.R`.

Part 1: Compare each linguistic domain to the overall cultural similarity

We fit a mixed effects model to compare the semantic alignment in a given domain to the overall cultural distance. The semantic alignment for the given domain is the dependent variable. There are random intercepts for language family and area pairs, and random slopes for overall cultural similarity by language family and by area. The `comparison_count` variable is added as a fixed effect. This null model is compared to a model with an additional fixed effect for the overall cultural similairty.

There are 21 linguistic domains with enough data. All correlations are positive and 11 are significant at the 0.05 level (adjusted for multiple comparisons).

The full results are in the file:

`../results/stats/wikipedia-main/Cor_LingAlignmentByDomains_vs_OverallCulturalSimilarity.csv`

Summary:

```
p1res
```

```
##                            Domain       Beta            p   Adjusted p sig
## 2                   Food and drink 0.29039152 3.842274e-08 8.068775e-07   *
## 6    Miscellaneous function words 0.31349670 9.672370e-08 2.031198e-06   *
## 9                        The body 0.23183657 8.711593e-07 1.829434e-05   *
## 13                        Animals 0.26483784 4.281952e-06 8.992099e-05   *
## 21                           Time 0.26708073 3.341177e-05 7.016471e-04   *
## 3      Agriculture and vegetation 0.21319270 4.954909e-05 1.040531e-03   *
## 16                    Modern world 0.15392213 2.860946e-04 6.007988e-03   *
## 14              The physical world 0.15530592 6.771587e-04 1.422033e-02   *
## 11               Spatial relations 0.11188738 1.323355e-03 2.779045e-02   *
## 20                        Kinship 0.25408132 1.332699e-03 2.798669e-02   *
## 7           Clothing and grooming 0.16478921 2.245060e-03 4.714625e-02   *
## 10               Sense perception 0.11168260 2.806430e-03 5.893504e-02
## 15 Social and political relations 0.10210872 6.603817e-03 1.386802e-01
## 1                       The house 0.10743767 1.485121e-02 3.118755e-01
## 18                       Quantity 0.13804241 1.691864e-02 3.552914e-01
## 8              Speech and language 0.11367507 3.209804e-02 6.740588e-01
## 19   Basic actions and technology 0.06996793 7.317704e-02 1.000000e+00
## 17                      Cognition 0.06413968 9.294337e-02 1.000000e+00
## 12            Emotions and values 0.06324463 9.437249e-02 1.000000e+00
## 5                      Possession 0.07833831 1.102507e-01 1.000000e+00
## 4                          Motion 0.05544251 2.537090e-01 1.000000e+00
```

Part 2: Compare each linguistic domain to the cultural similarity of each original D-PLACE domain

The method is the same as for part 1, except the cultural distance for a particular cultural domain is used instead of the overall cultural distance.

The full results are in the file:

`../results/stats/wikipedia-main/Cor_LingAlignmentByDomains_vs_DPlaceCulturalDomains.csv`

The graph below shows the mixed effects model coefficient estimate for the relationship between each linguistic domain and each cultural domain. Pink colours indicate positive correlations and blue colours indicate negative correlations. Stronger colours indicate stronger correlations. An asterisk indicates that the correlation is stronger than would be expected by chance, when adjusting the p-value for multiple comparisons.

The insert in the top left shows the distribution of Beta values.

The domains are clustered using higherarchical clustering. This is for visualisaiton and reflects similarity in the numeric relations, not history or conceptual hierarchies.

List of significant correlations (after adjusting p-value for multiple comparisons):

```
##                         Ling Domain Cult Domain      Beta   Adjusted p
## 68                         The body    Politics 0.2139638 1.292033e-03
## 155                         Kinship  Settlement 0.2785591 8.766279e-03
## 43    Miscellaneous function words  Settlement 0.2899807 7.022120e-07
## 83                Spatial relations  Settlement 0.1122428 1.431007e-03
## 59              Speech and language  Settlement 0.1915031 3.729177e-05
## 107              The physical world  Settlement 0.1321783 4.102261e-02
## 17      Agriculture and vegetation Subsistence 0.2513632 2.887202e-05
## 97                          Animals Subsistence 0.2942266 7.865336e-06
## 49            Clothing and grooming Subsistence 0.2484714 1.338589e-04
## 89              Emotions and values Subsistence 0.1524638 7.946118e-04
## 9                    Food and drink Subsistence 0.3005301 1.084517e-06
## 153                         Kinship Subsistence 0.2346825 4.160376e-03
## 41    Miscellaneous function words Subsistence 0.3353616 2.648483e-05
## 121                   Modern world Subsistence 0.1851981 7.434982e-06
## 137                       Quantity Subsistence 0.2454534 7.092645e-03
## 73                Sense perception Subsistence 0.1850017 6.280926e-04
## 113 Social and political relations Subsistence 0.1544007 3.637901e-05
## 81                Spatial relations Subsistence 0.1504141 6.897505e-04
## 57              Speech and language Subsistence 0.1991926 2.699664e-04
## 65                         The body Subsistence 0.2764446 5.291731e-08
## 105             The physical world Subsistence 0.2221189 1.157346e-04
## 161                           Time Subsistence 0.2921557 1.640226e-05
```

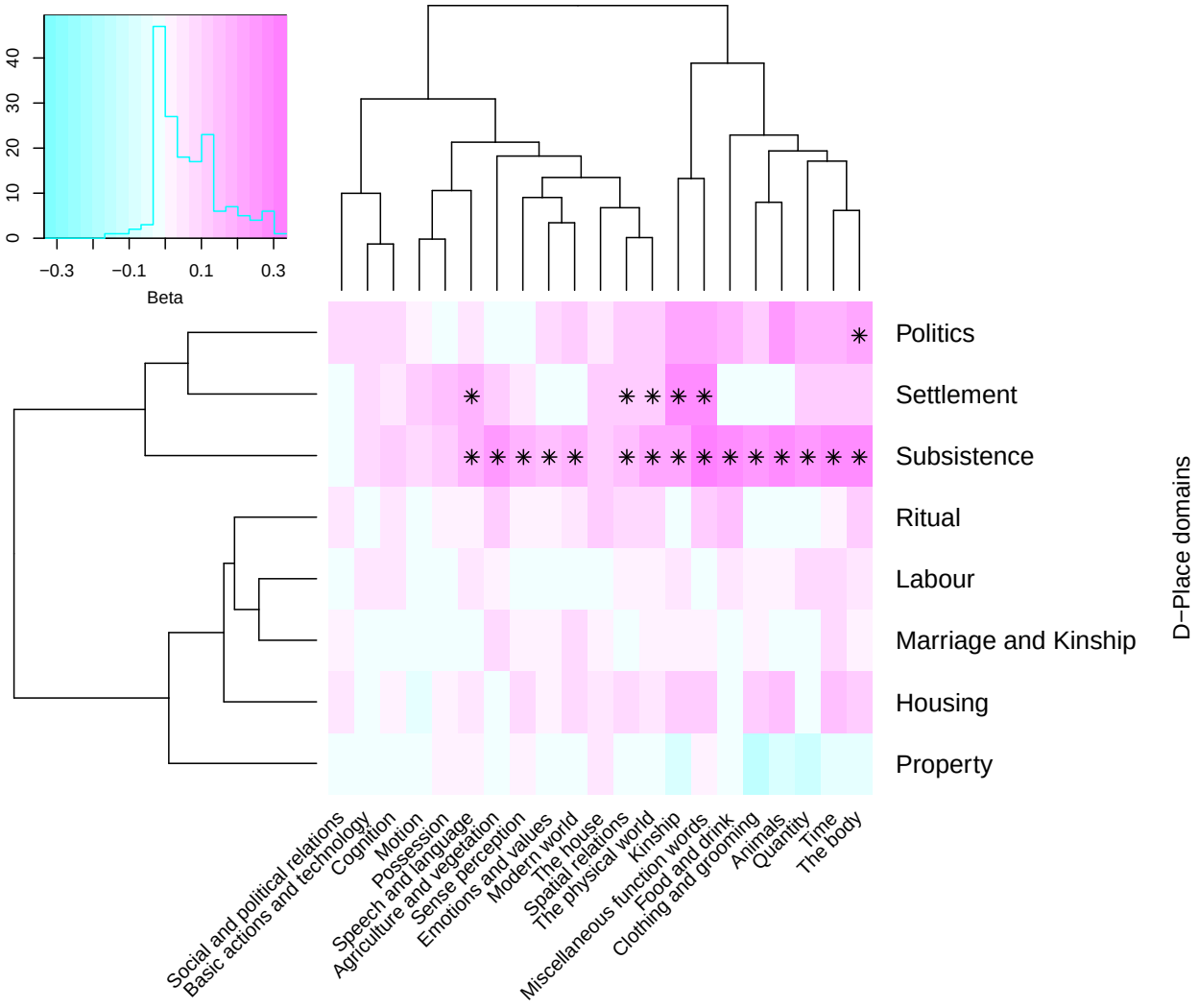

0
10
20
30
40
−0.3
−0.1
0.1
0.3
Beta
Politics
Settlement
Subsistence
Ritual
Labour
Marriage and Kinship
Housing
Property
D−Place domains
Social and political relations
Basic actions and technology
Cognition
Motion
Possession
Speech and language
Agriculture and vegetation
Sense perception
Emotions and values
Modern world
The house
Spatial relations
The physical world
Kinship
Miscellaneous function words
Food and drink
Clothing and grooming
Animals
Quantity
Time
The body
Concepticon domains

Figure 1:

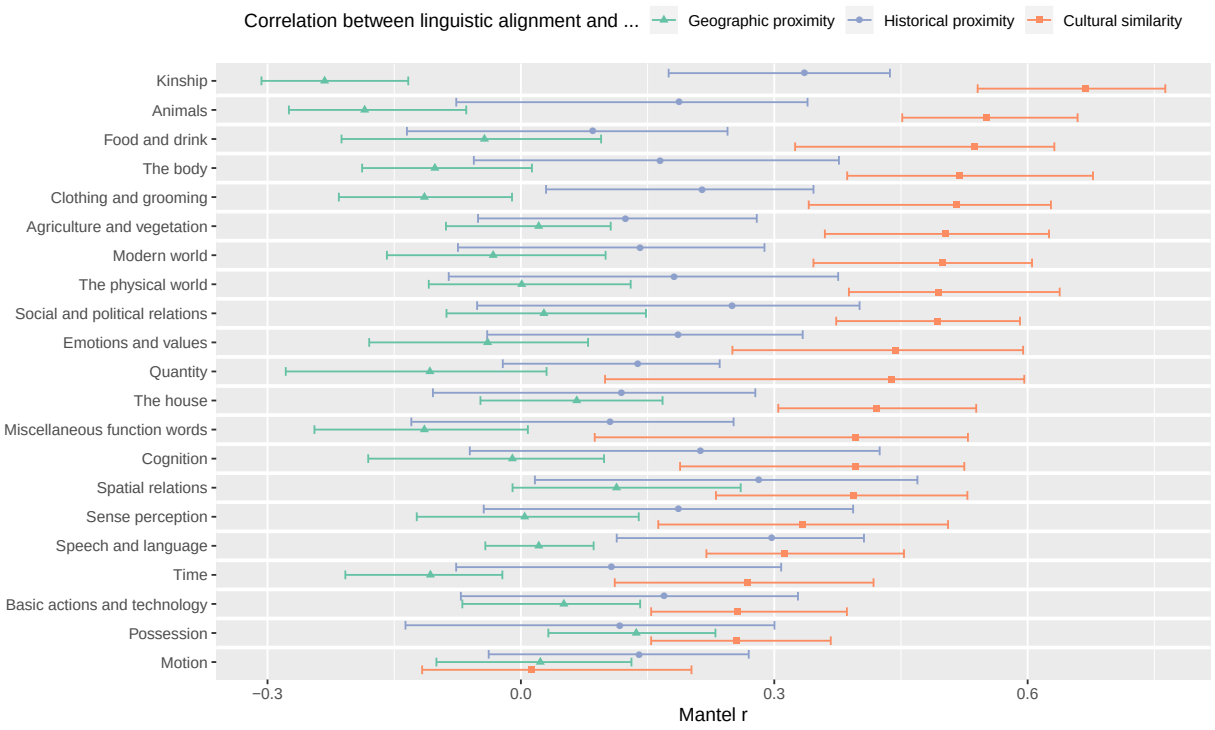


Figure 2:

Part 3: Compare each linguistic domain to the phylogenetic and geographic distance

This test compares each semantic alignment score to each of three target distances: the cultural distance, the historical distance and the geographic distance. We use a partial Mantel test (from the package `ecodist`) to estimate the strength of the relationship between the linguistic domain and the target distance, while controlling for the other two distances. The test uses 100,000 permutations.

The full results are in the file:

`Cor_LingAlignmentByDomains_vs_HistoricalAndGeographicalDistance.csv`

The graph below shows the results. Point estimates are the estimated Mantel R. The error bars show the 95% confidence intervals from the permutation test.

There appears to be a trade-off: The stronger the relationship with geographic distance, the weaker the relationship with cultural distance (r = -0.529, t = -2.72, df=19, p = 0.014). This does not hold for historical and cultural distance (r = 0.27, t = 1.22, df=19, p = 0.24).

Note that, after controlling for multiple comparisons, only 2 domains are significant:

```
##     domain comparison   mantelr     lower     upper   pval3 p.adjusted
## 37 Animals  lingVCult 0.5518312 0.4515039 0.6591382 0.00129    0.02709
## 58 Kinship  lingVCult 0.6687835 0.5407906 0.7629987 0.00012    0.00252
```

References

Bouckaert, Remco, Philippe Lemey, Michael Dunn, Simon J. Greenhill, Alexander V. Alekseyenko, Alexei J. Drummond, Russell D. Gray, Marc A. Suchard, and Quentin D. Atkinson (2012). Mapping the origins and expansion of the Indo-European language family. Science, 337(6097), 957-960.

van Buuren, S. & Groothuis-Oudshoorn, K. (2011). mice: Multivariate Imputation by Chained Equations in R. Journal of Statistical Software, 45(3), 1-67. URL https://www.jstatsoft.org/v45/i03/.

Castellano, S., & Balletto, E. (2002). Is the partial Mantel test inadequate?. Evolution, 56(9), 1871-1873.

Goslee, S.C. 2010. Correlation analysis of dissimilarity matrices. Plant Ecology 206(2):279-286.

Goslee, S.C. & Urban, D.L. (2007). The ecodist package for dissimilarity-based analysis of ecological data. Journal of Statistical Software 22(7):1-19.

Hammarstrom, H. , R. Forkel, M. Haspelmath (2018) clld/glottolog: Glottolog database 3.3, Jena: Max Planck Institute for the Science of Human History.

Harmon, L. J., & Glor, R. E. (2010). Poor statistical performance of the Mantel test in phylogenetic comparative analyses. Evolution: International Journal of Organic Evolution, 64(7), 2173-2178.

Jäger, G. (2018). Global-scale phylogenetic linguistic inference from lexical resources. Scientific data, 5, 180189.

Kirby, Kathryn R., Russell D. Gray, Simon J. Greenhill, Fiona M. Jordan, Stephanie Gomes-Ng, Hans-Jörg Bibiko, Damián E. Blasi, Carlos A. Botero, Claire Bowern, Carol R. Ember, Dan Leehr, Bobbi S. Low, Joe McCarter, William Divale, and Michael C. Gavin. (2016). D-PLACE: A Global Database of Cultural, Linguistic and Environmental Diversity. PLoS ONE, 11(7): e0158391.

Legendre, P. (2000). Comparison of permutation methods for the partial correlation and partial Mantel tests. Journal of Statistical Computation and Simulation, 67(1), 37-73.

Lichstein, J. W. (2007). Multiple regression on distance matrices: a multivariate spatial analysis tool. Plant Ecology, 188(2), 117-131.

Mantel, N. (1967). The detection of disease clustering and a generalized regression approach. Cancer research, 27(2 Part 1), 209-220.

Murdock, G. P., R. Textor, H. Barry, III, D. R. White, J. P. Gray, and W. T. Divale. 1999. Ethnographic Atlas. World Cultures 10:24-136 (codebook)

Nichols, J., Witzlack-Makarevich, A. & Bickel, B. (2013), The AUTOTYP genealogy and geography database: 2013 release, http://www.spw.uzh.ch/autotyp/.

Smouse, P.E., Long, J.C. & Sokal, R.R. (1986). Multiple regression and correlation extensions of the Mantel test of matrix correspondence. Systematic Zoology 35:62 7-632.

Wichmann, Søren, Eric W. Holman, and Cecil H. Brown (eds.). 2018. The ASJP Database (version 17).