Cultural distances: Wikipedia data

Contents

Introduction

We compare cultural distances between socieites with linguistic similarities 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) is used as a random effect. That means that the model can capture the likelihood that two languages from the Indo-European language family will be more similar to each other than two languages from different language families. The same is done with geographic area according to Autotyp.

The second test controls for history using distances from a phylogenetic tree. The tree comes from Bouckaert et al. (2012). Patristic distances between languages are used as a measure of historical distance between societies in a Mantel test. Note that the Mantel test assumes a strict distance metric, which is not necessarily the case with this data, but there are few other ways to deal with continuous pairwise distances.

Load libraries

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

Parameters:

```
datasetName = "wikipedia-main"
lingDistancesFile = "../data/FAIR/main-data-nel-wikipedia-k100-by-language-pair.csv"
lingDistancesByDomainFile = "../results/EA_distances/wikipedia_All_Domains_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")
cultLangs = unique(c(cult$Var1,cult$Var2))
```

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

```
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
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$family1 = l[match(ling$l1, l$iso2),]$family
ling$family2 = l[match(ling$l2, l$iso2),]$family
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
## 3  ab he      0.07566733 Abkhaz-Adyge:Afro-Asiatic
## 5  ab zh      0.02225169 Abkhaz-Adyge:Sino-Tibetan
## 38 ba ab      0.07330162       Abkhaz-Adyge:Turkic
## 42 ba cv      0.18792509             Turkic:Turkic
## 44 ba he      0.17701891       Afro-Asiatic:Turkic
## 45 ba ja      0.13254527            Japonic:Turkic
```

And the same is true for area:

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

```
##       l1 l2 local_alignment                area.group
## 2527 xal ja      0.02832668   Inner Asia:N Coast Asia
## 2530 xal ko      0.04879589   Inner Asia:N Coast Asia
## 2532 xal ml      0.07500293          Indic:Inner Asia
## 2533 xal ta      0.06311429          Indic:Inner Asia
## 2534 xal te      0.05080388          Indic:Inner Asia
## 2535 xal zh      0.02895876 Inner Asia:Southeast Asia
```

Number of observations:

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

```
## [1] 733
```

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

```
## [1] 40
```

```
# 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          5            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 Linguistic similaritys from cultural distances, with random intercept for family and area and random slope for cultural distance for family and area.

We compare a null model to a model with a fixed effect for cultural distance, with random intercepts for family and area, and random slopes for cultural distance by both.

```
m0 = lmer(
  rho.center ~ 1 +
    (1 + cult.dist.center | family.group) +
    (1 + cult.dist.center | area.group),
  data = ling
)
m1 = lmer(
  rho.center ~ 1 +
    cult.dist.center +
    (1 + cult.dist.center | family.group) +
    (1 + cult.dist.center | area.group),
  data = ling
)
anova(m0,m1)
```

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

```
## Data: ling
## Models:
## m0: rho.center ~ 1 + (1 + cult.dist.center | family.group) + (1 +
## m0:     cult.dist.center | area.group)
## m1: rho.center ~ 1 + cult.dist.center + (1 + cult.dist.center | family.group) +
## m1:     (1 + cult.dist.center | area.group)
##    Df    AIC    BIC  logLik deviance  Chisq Chi Df Pr(>Chisq)
## m0  8 1658.0 1694.7 -820.97   1642.0
## m1  9 1641.8 1683.2 -811.93   1623.8 18.091      1  2.106e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Cultural distance is significantly correlated with Linguistic similarity. Here are the model estimates:

```
summary(m1)
```

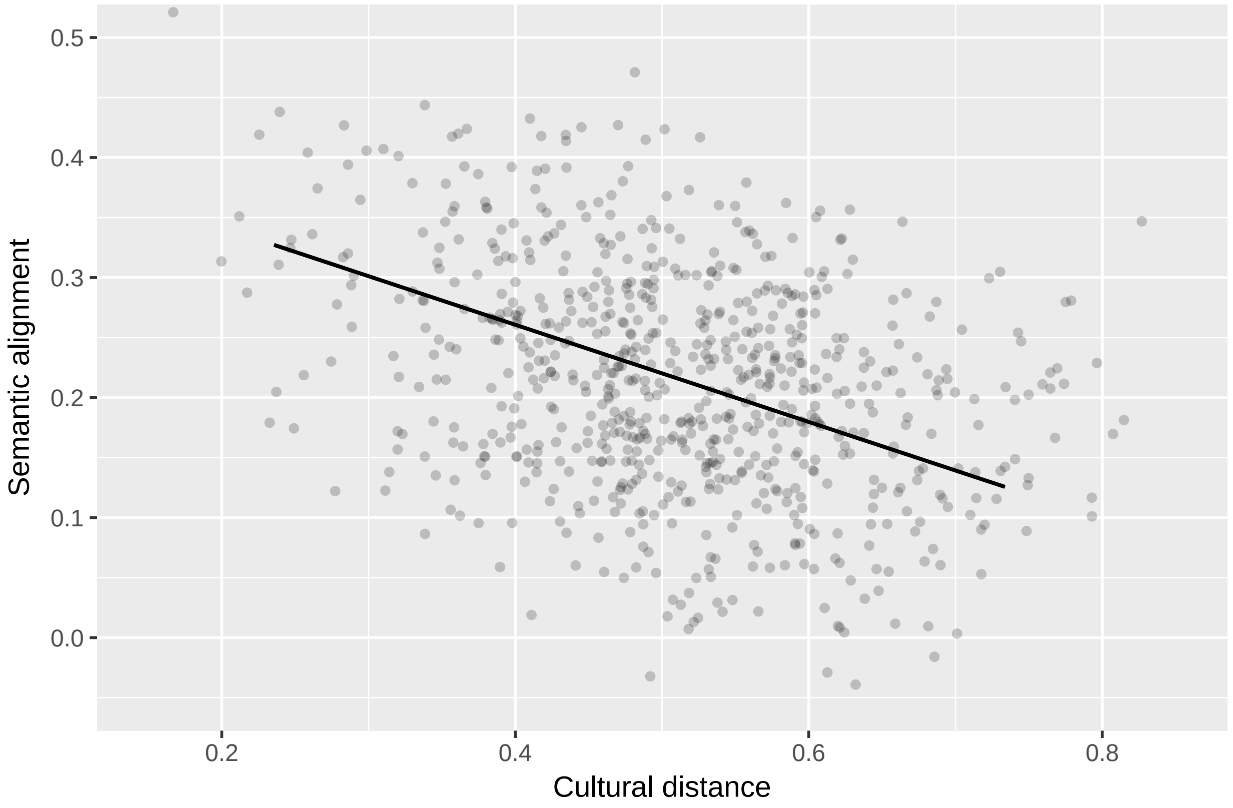
```
## Linear mixed model fit by REML ['lmerMod']
## Formula:
## rho.center ~ 1 + cult.dist.center + (1 + cult.dist.center | family.group) +
##     (1 + cult.dist.center | area.group)
##    Data: ling
##
## REML criterion at convergence: 1630.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.6291 -0.6811  0.0713  0.6658  4.6028
##
## Random effects:
##  Groups       Name             Variance  Std.Dev.  Corr
##  family.group (Intercept)      3.964e-01 6.296e-01
##               cult.dist.center 2.993e-02 1.730e-01 0.03
##  area.group   (Intercept)      0.000e+00 0.000e+00
##               cult.dist.center 9.586e-15 9.791e-08  NaN
##  Residual                      4.697e-01 6.853e-01
## Number of obs: 733, groups:  family.group, 48; area.group, 20
##
## Fixed effects:
##                  Estimate Std. Error t value
## (Intercept)      -0.60832    0.10834  -5.615
## cult.dist.center -0.30894    0.05779  -5.346
##
## Correlation of Fixed Effects:
##             (Intr)
## clt.dst.cnt -0.142
```

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

```
gx = sjp.lmer(m1,'pred','cult.dist.center',
              prnt.plot = F)
```

```
## Warning: package 'bindrcpp' was built under R version 3.3.2
```

```
gx$plot$data$y = gx$plot$data$y *
  attr(ling$rho.center,"scaled:scale") +
  attr(ling$rho.center,"scaled:center")
gx$plot$data$resp.y = gx$plot$data$resp.y *
  attr(ling$rho.center,"scaled:scale") +
  attr(ling$rho.center,"scaled:center")
gx$plot$data$x = gx$plot$data$x *
  cdc.s +cdc.c
gx = gx$plot + coord_cartesian(ylim=c(-0.05,0.5),
                          xlim=c(0.15,0.85)) +
  xlab("Cultural distance") +
  ylab("Semantic alignment") +
  ggtitle("")
gx
```



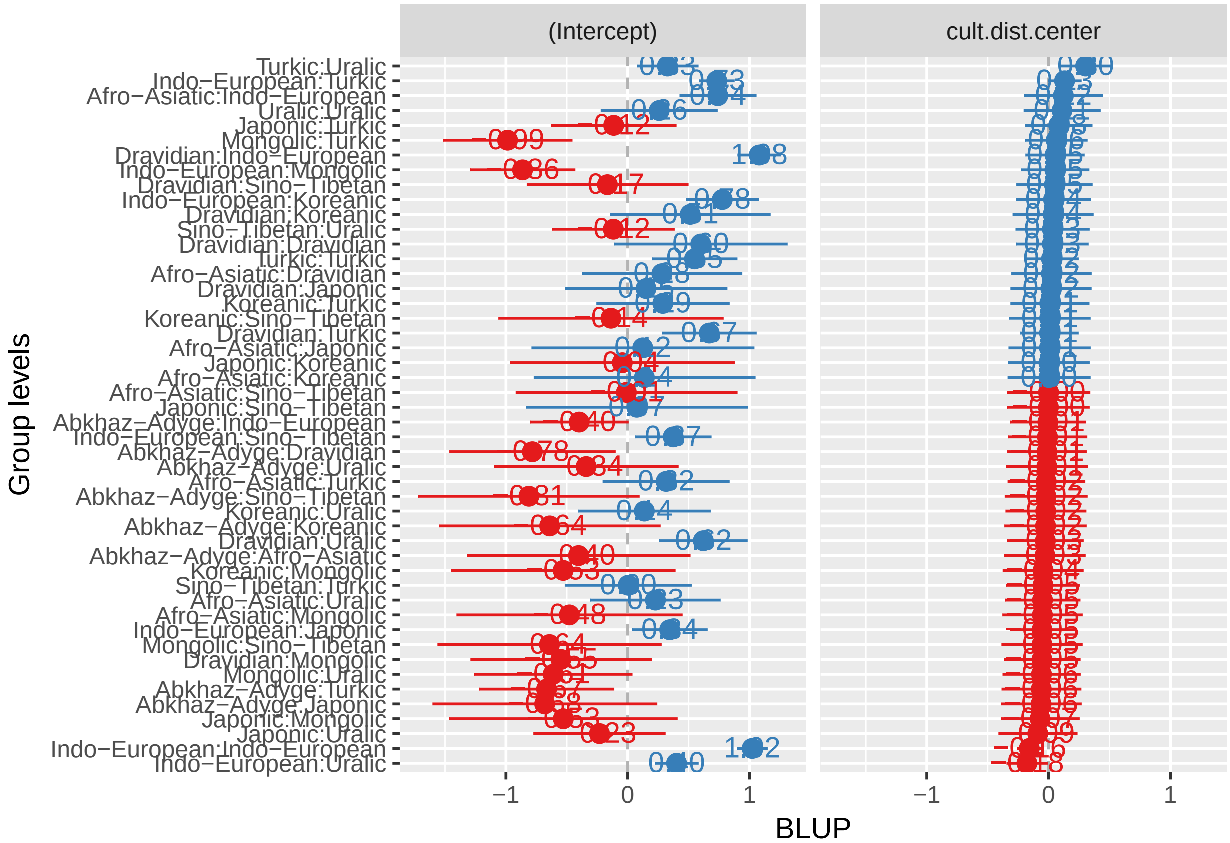
```
pdf(paste0("../results/stats/",datasetName,"/CulturalDistance_Rho_Graph.pdf"),
    height=2.5, width=2.5)
gx
dev.off()
```

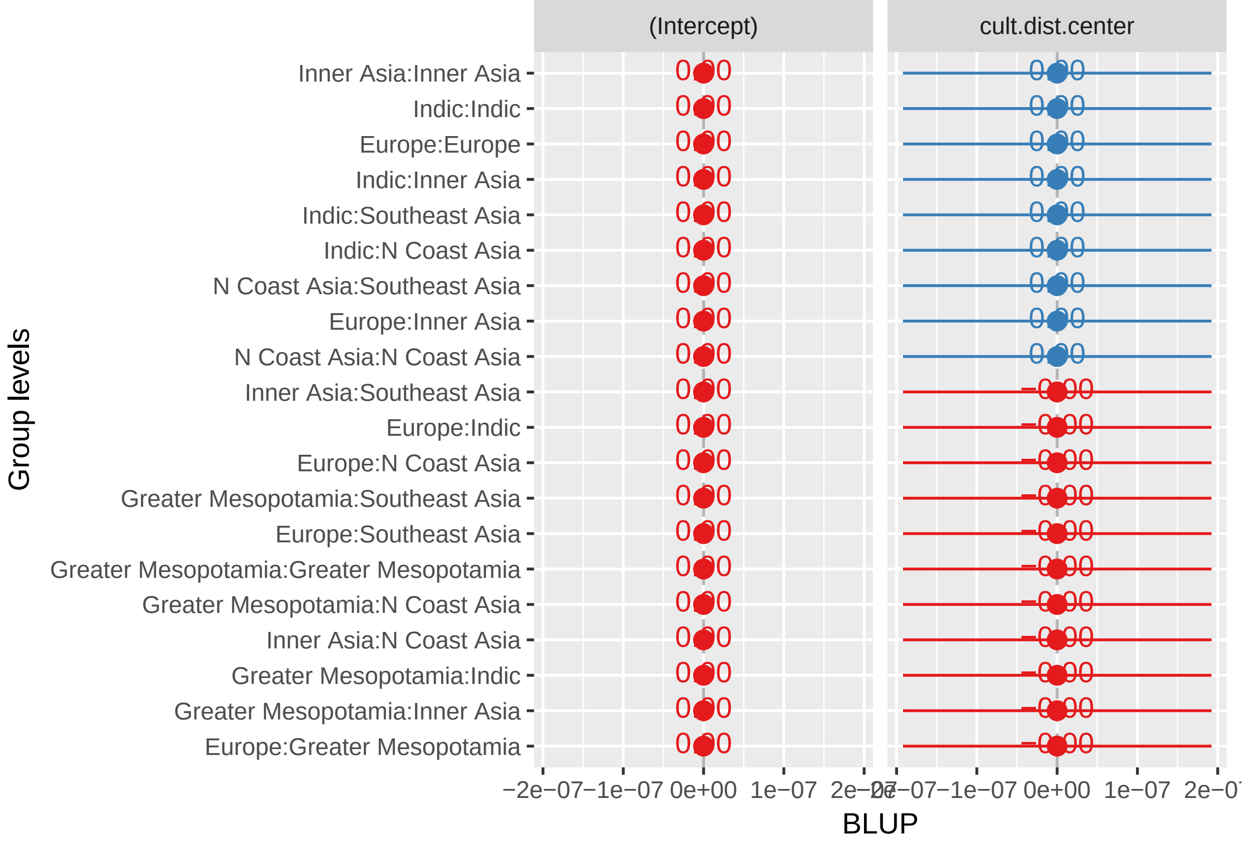
```
## pdf
##   2
```

Plot the random effects:

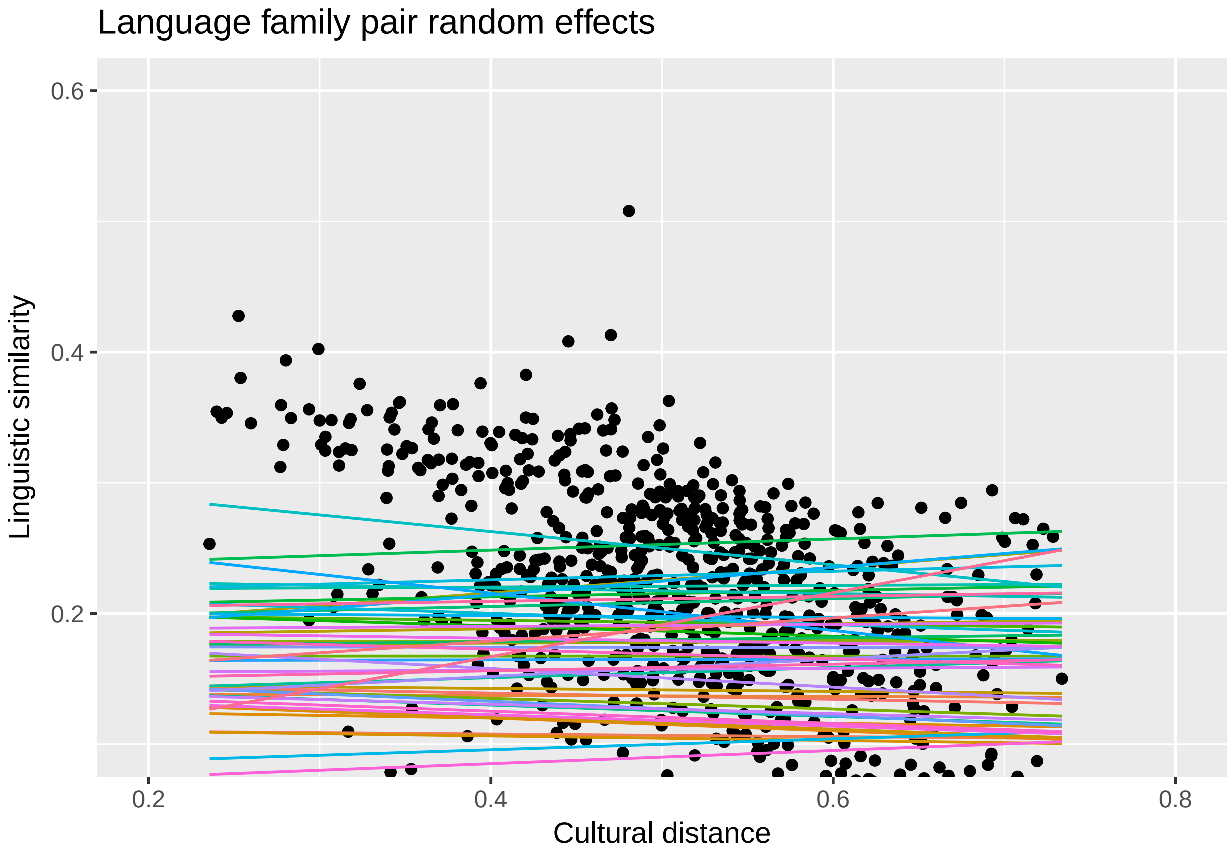
```
sjp.lmer(m1,'re', sort.est = "cult.dist.center")
```

```
## Plotting random effects...
## Plotting random effects...
```

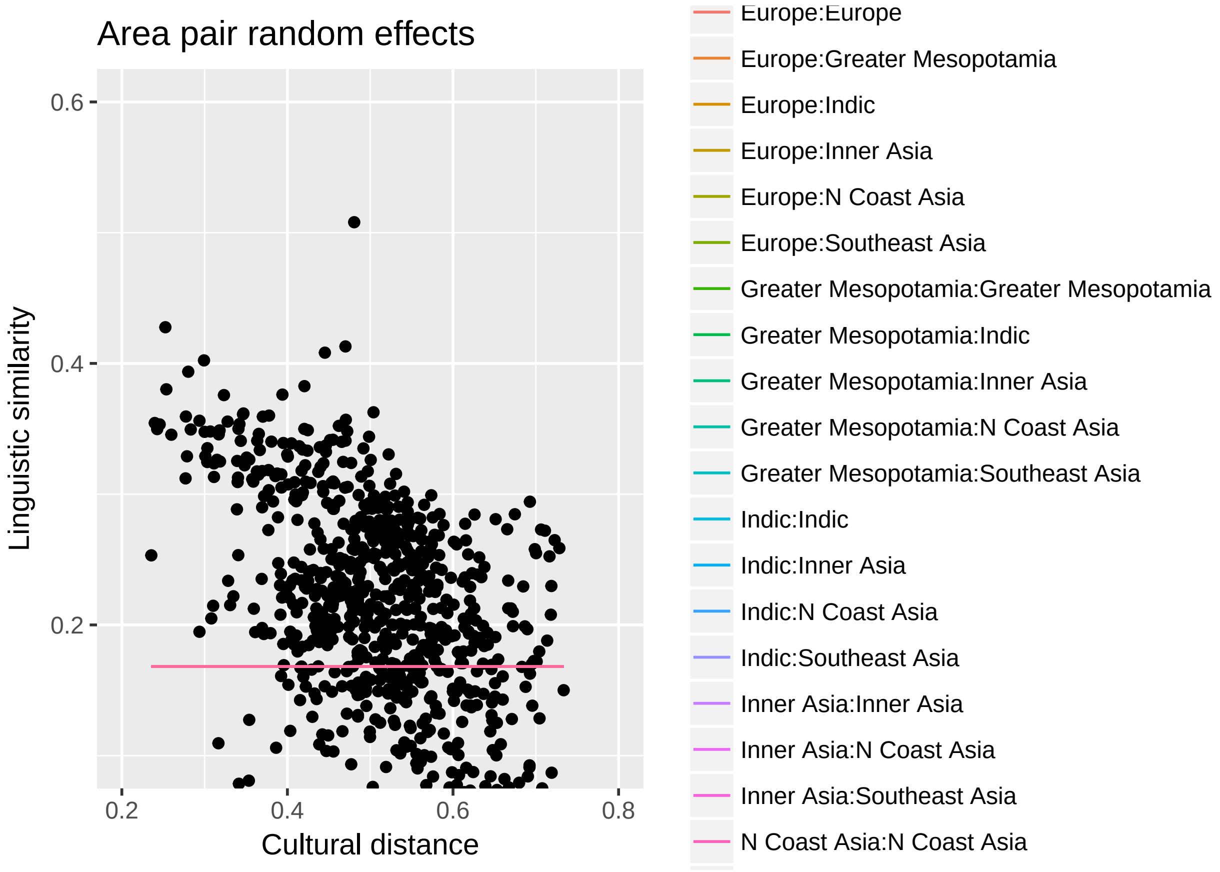

(Intercept)
cult.dist.center
Group levels
Turkic:Uralic
Indo−European:Turkic
Afro−Asiatic:Indo−European
Uralic:Uralic
Japonic:Turkic
Mongolic:Turkic
Dravidian:Indo−European
Indo−European:Mongolic
Dravidian:Sino−Tibetan
Indo−European:Koreanic
Dravidian:Koreanic
Sino−Tibetan:Uralic
Dravidian:Dravidian
Turkic:Turkic
Afro−Asiatic:Dravidian
Dravidian:Japonic
Koreanic:Turkic
Koreanic:Sino−Tibetan
Dravidian:Turkic
Afro−Asiatic:Japonic
Japonic:Koreanic
Afro−Asiatic:Koreanic
Afro−Asiatic:Sino−Tibetan
Japonic:Sino−Tibetan
Abkhaz−Adyge:Indo−European
Indo−European:Sino−Tibetan
Abkhaz−Adyge:Dravidian
Abkhaz−Adyge:Uralic
Afro−Asiatic:Turkic
Abkhaz−Adyge:Sino−Tibetan
Koreanic:Uralic
Abkhaz−Adyge:Koreanic
Dravidian:Uralic
Abkhaz−Adyge:Afro−Asiatic
Koreanic:Mongolic
Sino−Tibetan:Turkic
Afro−Asiatic:Uralic
Afro−Asiatic:Mongolic
Indo−European:Japonic
Mongolic:Sino−Tibetan
Dravidian:Mongolic
Mongolic:Uralic
Abkhaz−Adyge:Turkic
Abkhaz−Adyge:Japonic
Japonic:Mongolic
Japonic:Uralic
Indo−European:Indo−European
Indo−European:Uralic
1.08
1.02
−1
0
1
−1
0
1
BLUP


(Intercept)
cult.dist.center
Group levels
Inner Asia:Inner Asia
Indic:Indic
Europe:Europe
Indic:Inner Asia
Indic:Southeast Asia
Indic:N Coast Asia
N Coast Asia:Southeast Asia
Europe:Inner Asia
N Coast Asia:N Coast Asia
Inner Asia:Southeast Asia
Europe:Indic
Europe:N Coast Asia
Greater Mesopotamia:Southeast Asia
Europe:Southeast Asia
Greater Mesopotamia:Greater Mesopotamia
Greater Mesopotamia:N Coast Asia
Inner Asia:N Coast Asia
Greater Mesopotamia:Indic
Greater Mesopotamia:Inner Asia
Europe:Greater Mesopotamia
0.00
−0.00
−2e−07
−1e−07
0e+00
1e−07
2e−07
−2e−07
−1e−07
0e+00
1e−07
2e−0
BLUP

```
px = sjp.lmer(m1,'rs.ri', prnt.plot = F)
dx = px$plot[[1]]$data
dx$x = dx$x * cdc.s + cdc.c
dx$y = dx$y *
  attr(ling$rho.center,"scaled:scale") +
  attr(ling$rho.center,"scaled:center")
ggplot(dx,aes(x,y)) +
  geom_point(data=ling,
           mapping=aes(x=as.numeric(cult.dist),
                       y=as.numeric(local_alignment))) +
  geom_line(mapping = aes(colour=grp)) +
  xlab("Cultural distance")+
  ylab("Linguistic similarity") +
  ggtitle("Language family pair random effects") +
  coord_cartesian(ylim=c(0.1,0.6),
                xlim=c(0.2,0.8)) +
  theme(legend.position = "none")
```



```
dx = px$plot[[2]]$data
dx$x = dx$x * cdc.s + cdc.c
dx$y = dx$y *
  attr(ling$rho.center,"scaled:scale") +
  attr(ling$rho.center,"scaled:center")
ggplot(dx,aes(x,y)) +
  geom_point(data=ling,
           mapping=aes(x=as.numeric(cult.dist),
                         y=as.numeric(local_alignment))) +
  geom_line(mapping = aes(colour=grp)) +
  xlab("Cultural distance")+
  ylab("Linguistic similarity") +
  ggtitle("Area pair random effects") +
  coord_cartesian(ylim=c(0.1,0.6),
                  xlim=c(0.2,0.8))
```



Note that the random slopes for area are set to 0. We can check whether taking language area out makes a difference:

```
m0b = lmer(
  rho.center ~ 1 +
    (1 + cult.dist.center | family.group),
  data = ling
)
m1b = lmer(
  rho.center ~ 1 +
    cult.dist.center +
    (1 + cult.dist.center | family.group),
  data = ling
)
anova(m0b,m1b)
```

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

```
## Data: ling
## Models:
## m0b: rho.center ~ 1 + (1 + cult.dist.center | family.group)
## m1b: rho.center ~ 1 + cult.dist.center + (1 + cult.dist.center | family.group)
##     Df    AIC    BIC  logLik deviance  Chisq Chi Df Pr(>Chisq)    
## m0b  5 1652.0 1674.9 -820.97   1642.0                             
## m1b  6 1635.8 1663.4 -811.93   1623.8 18.091      1  2.106e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(m1b)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: 
## rho.center ~ 1 + cult.dist.center + (1 + cult.dist.center | family.group)
##    Data: ling
## 
## REML criterion at convergence: 1630.4
## 
## Scaled residuals: 
##     Min      1Q  Median      3Q     Max 
## -2.6291 -0.6811  0.0713  0.6658  4.6028 
## 
## Random effects:
##  Groups       Name             Variance Std.Dev. Corr
##  family.group (Intercept)      0.39642  0.6296       
##               cult.dist.center 0.02993  0.1730   0.03
##  Residual                      0.46969  0.6853       
## Number of obs: 733, groups:  family.group, 48
## 
## Fixed effects:
##                  Estimate Std. Error t value
## (Intercept)      -0.60832    0.10834  -5.615
## cult.dist.center -0.30894    0.05779  -5.346
## 
## Correlation of Fixed Effects:
##             (Intr)
## clt.dst.cnt -0.142
```

The model is numerically almost exactly the same.

Tests within domains

Load data

Load distances for specific domains and match up to language family and area:

```
ling.dom = read.csv(
  lingDistancesByDomainFile,
  stringsAsFactors = F)

ling.dom = ling.dom[!is.na(ling.dom$cult.dist),]

ling.dom = ling.dom[(ling.dom$l1 %in% cultisos) &
                      (ling.dom$l2 %in% cultisos),]
```

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

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

Match family and area data:

```
ling.dom$family1 = l[match(ling.dom$l1, l$iso2),]$family
ling.dom$family2 = l[match(ling.dom$l2, l$iso2),]$family
ling.dom$area1 = l[match(ling.dom$l1, l$iso2),]$autotyp.area
ling.dom$area2 = l[match(ling.dom$l2, l$iso2),]$autotyp.area


# Paste language family names together,
# but order shouldn't matter, so sort first
fgroup = cbind(ling.dom$family1,ling.dom$family2)
fgroup = apply(fgroup,1,sort)
ling.dom$family.group = apply(fgroup,2,paste,collapse=":")

agroup = cbind(ling.dom$area1,ling.dom$area2)
agroup = apply(agroup,1,sort)
ling.dom$area.group = apply(agroup,2,paste,collapse=":")
```

Center the data:

```
ling.dom$cult.dist.center = scale(ling.dom$cult.dist)
ling.dom$rho.center = scale(ling.dom$local_alignment)
```

LMER models

Test whether random slopes are warraneted for family:

```
mD0 = lmer(
  rho.center ~ 1 +
    (1 | family.group) +
    (1 | area.group) +
    (1 | imputed_semantic_domain),
  data = ling.dom)
mD1 = lmer(
  rho.center ~ 1 +
    (1 | family.group) +
    (0 + cult.dist.center | family.group) +
    (1 | area.group) +
    (1 | imputed_semantic_domain),
  data = ling.dom)
mD2 = lmer(
  rho.center ~ 1 +
    (1 + cult.dist.center | family.group) +
    (1 | area.group) +
    (1 | imputed_semantic_domain),
  data = ling.dom)
anova(mD0,mD1,mD2)
```

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

## Data: ling.dom
## Models:
## mD0: rho.center ~ 1 + (1 | family.group) + (1 | area.group) + (1 |
## mD0:     imputed_semantic_domain)
## mD1: rho.center ~ 1 + (1 | family.group) + (0 + cult.dist.center |
## mD1:     family.group) + (1 | area.group) + (1 | imputed_semantic_domain)
## mD2: rho.center ~ 1 + (1 + cult.dist.center | family.group) + (1 |
## mD2:     area.group) + (1 | imputed_semantic_domain)
##     Df   AIC   BIC  logLik deviance   Chisq Chi Df Pr(>Chisq)
## mD0  5 12830 12863 -6409.8    12820
## mD1  6 12592 12632 -6289.7    12580 240.163      1    < 2e-16 ***
## mD2  7 12588 12635 -6287.1    12574   5.322      1    0.02106 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Random slopes (and the correlation coefficient) for family improves the fit of the model.

Test the same for area:

```
mD3 = lmer(
  rho.center ~ 1 +
    (1 + cult.dist.center | family.group) +
    (1 | area.group) +
    (0 + cult.dist.center | area.group) +
    (1 | imputed_semantic_domain),
  data = ling.dom)
mD4 = lmer(
  rho.center ~ 1 +
    (1 + cult.dist.center | family.group) +
    (1 + cult.dist.center | area.group) +
    (1 | imputed_semantic_domain),
  data = ling.dom)
anova(mD2,mD3,mD4)
```

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

## Data: ling.dom
## Models:
## mD2: rho.center ~ 1 + (1 + cult.dist.center | family.group) + (1 |
## mD2:     area.group) + (1 | imputed_semantic_domain)
## mD3: rho.center ~ 1 + (1 + cult.dist.center | family.group) + (1 |
## mD3:     area.group) + (0 + cult.dist.center | area.group) + (1 |
## mD3:     imputed_semantic_domain)
## mD4: rho.center ~ 1 + (1 + cult.dist.center | family.group) + (1 +
## mD4:     cult.dist.center | area.group) + (1 | imputed_semantic_domain)
##     Df   AIC   BIC  logLik deviance   Chisq Chi Df Pr(>Chisq)    
## mD2  7 12588 12635 -6287.1    12574                              
## mD3  8 12572 12625 -6277.9    12556 18.4041      1  1.787e-05 ***
## mD4  9 12569 12629 -6275.3    12551  5.1295      1    0.02352 *  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Random slopes for area improve the fit of the model.

Test random slopes for domain:

```r
mdom1 = lmer(
  rho.center ~ 1 +
    (1 +cult.dist.center| family.group) +
    (1 +cult.dist.center| area.group) +
    (1 | imputed_semantic_domain),
  data = ling.dom)
mdom2 = lmer(
  rho.center ~ 1 +
    (1 +cult.dist.center| family.group) +
    (1 +cult.dist.center| area.group) +
    (1 + cult.dist.center| imputed_semantic_domain),
  data = ling.dom)
anova(mdom1,mdom2)
```

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

```
## Data: ling.dom
## Models:
## mdom1: rho.center ~ 1 + (1 + cult.dist.center | family.group) + (1 + 
## mdom1:     cult.dist.center | area.group) + (1 | imputed_semantic_domain)
## mdom2: rho.center ~ 1 + (1 + cult.dist.center | family.group) + (1 + 
## mdom2:     cult.dist.center | area.group) + (1 + cult.dist.center | 
## mdom2:     imputed_semantic_domain)
##       Df   AIC   BIC  logLik deviance  Chisq Chi Df Pr(>Chisq)    
## mdom1  9 12569 12629 -6275.3    12551                             
## mdom2 11 12527 12600 -6252.4    12505 45.851      2  1.105e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Random slope for domains significantly improves model.

Now we test the main effect of cultural distance:

```r
mD5 = lmer(
  rho.center ~ 1 +
    (1 + cult.dist.center | family.group) +
    (1 + cult.dist.center| area.group) +
    (1 + cult.dist.center | imputed_semantic_domain),
  data = ling.dom)
mD6 = update(mD5, ~.+cult.dist.center)
anova(mD5,mD6)
```

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

```
## Data: ling.dom
## Models:
```

```
## mD5: rho.center ~ 1 + (1 + cult.dist.center | family.group) + (1 +
## mD5:     cult.dist.center | area.group) + (1 + cult.dist.center |
## mD5:     imputed_semantic_domain)
## mD6: rho.center ~ (1 + cult.dist.center | family.group) + (1 + cult.dist.center |
## mD6:     area.group) + (1 + cult.dist.center | imputed_semantic_domain) +
## mD6:     cult.dist.center
##     Df   AIC   BIC  logLik deviance  Chisq Chi Df Pr(>Chisq)
## mD5 11 12527 12600 -6252.4    12505
## mD6 12 12529 12609 -6252.3    12505 0.1742      1     0.6764
```

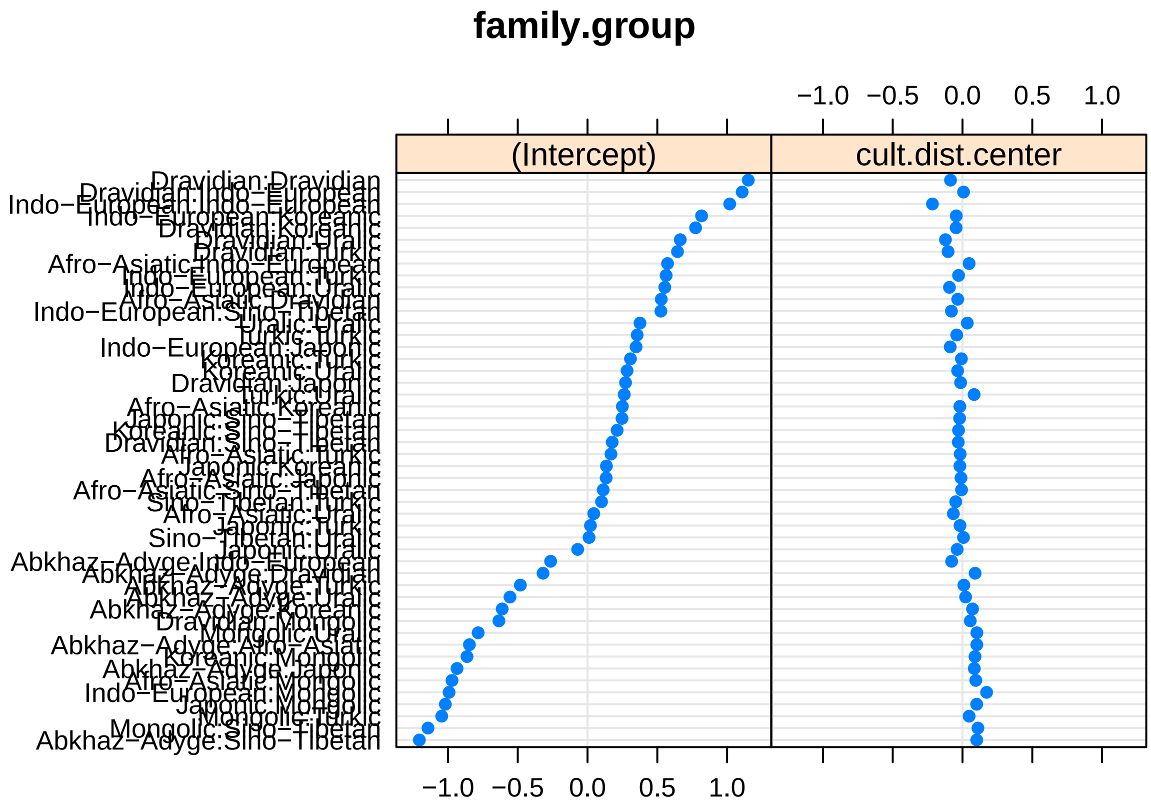
Summary of the final model, with random effects plot:

```
summary(mD6)
```

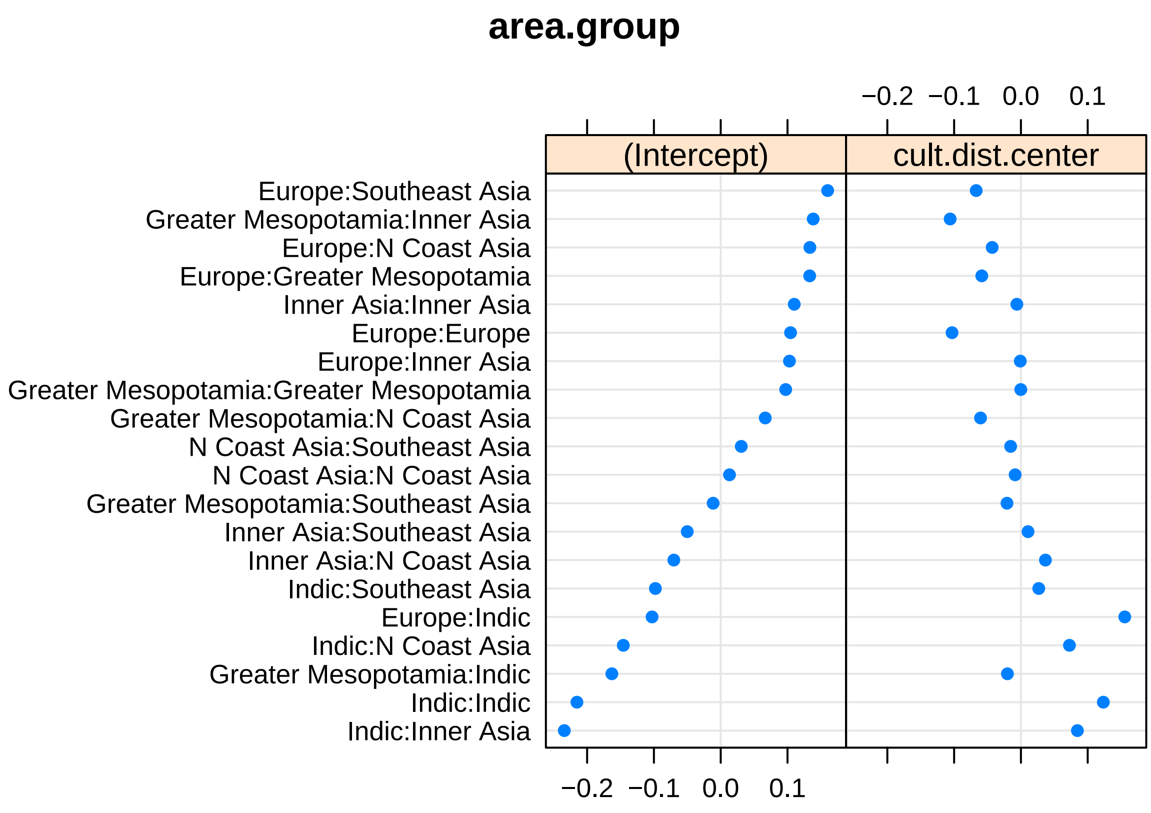
```
## Linear mixed model fit by REML ['lmerMod']
## Formula:
## rho.center ~ (1 + cult.dist.center | family.group) + (1 + cult.dist.center |
##     area.group) + (1 + cult.dist.center | imputed_semantic_domain) +
##     cult.dist.center
##    Data: ling.dom
##
## REML criterion at convergence: 12510.8
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -5.0342 -0.6137  0.0268  0.6099  3.8371
##
## Random effects:
##  Groups                  Name             Variance Std.Dev. Corr
##  family.group            (Intercept)      0.444533 0.66673
##                          cult.dist.center 0.011070 0.10521  -0.60
##  area.group              (Intercept)      0.024293 0.15586
##                          cult.dist.center 0.008002 0.08945  -0.67
##  imputed_semantic_domain (Intercept)      0.241060 0.49098
##                          cult.dist.center 0.007321 0.08556  -0.71
##  Residual                                 0.470128 0.68566
## Number of obs: 5858, groups:
## family.group, 48; area.group, 20; imputed_semantic_domain, 8
##
## Fixed effects:
##                  Estimate Std. Error t value
## (Intercept)      -0.63238    0.20317  -3.113
## cult.dist.center -0.01869    0.04463  -0.419
##
## Correlation of Fixed Effects:
##             (Intr)
## clt.dst.cnt -0.576
```

```
dotplot(ranef(mD6))
```

```
## $family.group
```

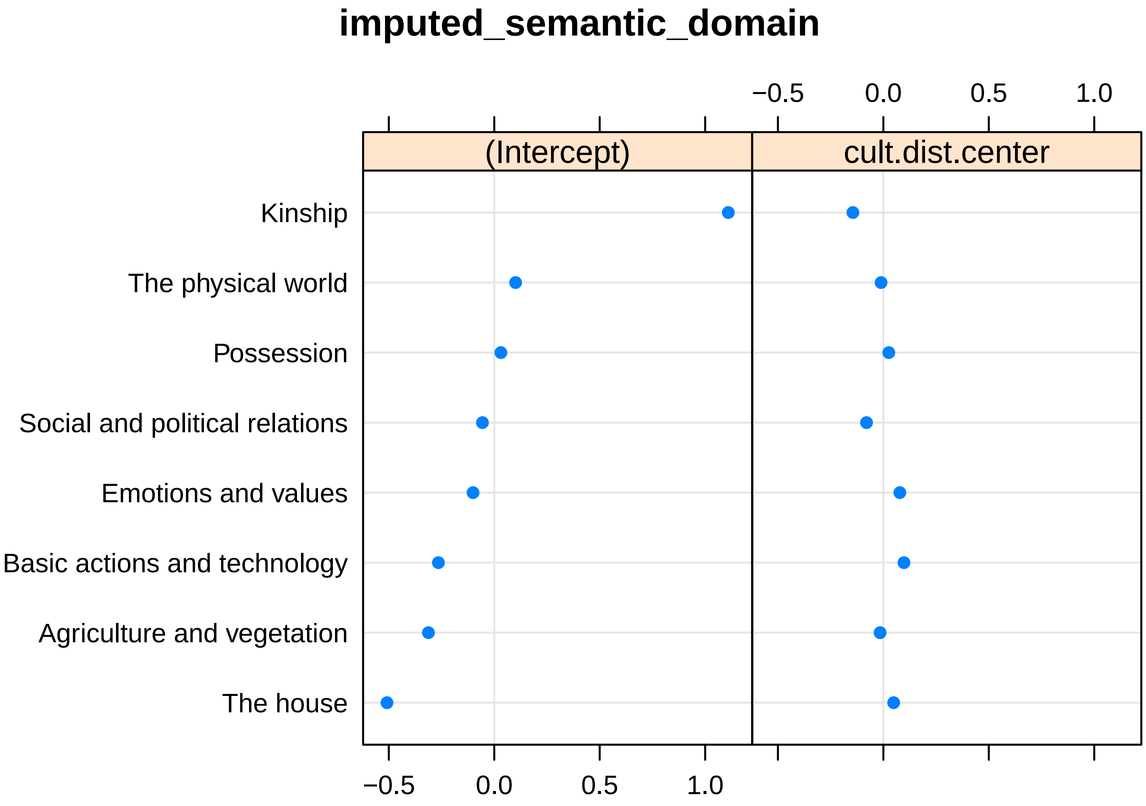

family.group
(Intercept)
cult.dist.center
−1.0
−0.5
0.0
0.5
1.0
Dravidian:Dravidian
Dravidian:Indo−European
Indo−European:Indo−European
Indo−European:Koreanic
Dravidian:Koreanic
Dravidian:Uralic
Dravidian:Turkic
Afro−Asiatic:Indo−European
Indo−European:Turkic
Indo−European:Uralic
Afro−Asiatic:Dravidian
Indo−European:Sino−Tibetan
Uralic:Uralic
Turkic:Turkic
Indo−European:Japonic
Koreanic:Turkic
Koreanic:Uralic
Dravidian:Japonic
Turkic:Uralic
Afro−Asiatic:Koreanic
Japonic:Sino−Tibetan
Koreanic:Sino−Tibetan
Dravidian:Sino−Tibetan
Afro−Asiatic:Turkic
Japonic:Koreanic
Afro−Asiatic:Japonic
Afro−Asiatic:Sino−Tibetan
Sino−Tibetan:Turkic
Afro−Asiatic:Uralic
Japonic:Turkic
Sino−Tibetan:Uralic
Japonic:Uralic
Abkhaz−Adyge:Indo−European
Abkhaz−Adyge:Dravidian
Abkhaz−Adyge:Turkic
Abkhaz−Adyge:Uralic
Abkhaz−Adyge:Koreanic
Dravidian:Mongolic
Mongolic:Uralic
Abkhaz−Adyge:Afro−Asiatic
Koreanic:Mongolic
Abkhaz−Adyge:Japonic
Afro−Asiatic:Mongolic
Indo−European:Mongolic
Japonic:Mongolic
Mongolic:Turkic
Mongolic:Sino−Tibetan
Abkhaz−Adyge:Sino−Tibetan

```
##
## $area.group
```


area.group
(Intercept)
cult.dist.center
−0.2
−0.1
0.0
0.1
Europe:Southeast Asia
Greater Mesopotamia:Inner Asia
Europe:N Coast Asia
Europe:Greater Mesopotamia
Inner Asia:Inner Asia
Europe:Europe
Europe:Inner Asia
Greater Mesopotamia:Greater Mesopotamia
Greater Mesopotamia:N Coast Asia
N Coast Asia:Southeast Asia
N Coast Asia:N Coast Asia
Greater Mesopotamia:Southeast Asia
Inner Asia:Southeast Asia
Inner Asia:N Coast Asia
Indic:Southeast Asia
Europe:Indic
Indic:N Coast Asia
Greater Mesopotamia:Indic
Indic:Indic
Indic:Inner Asia

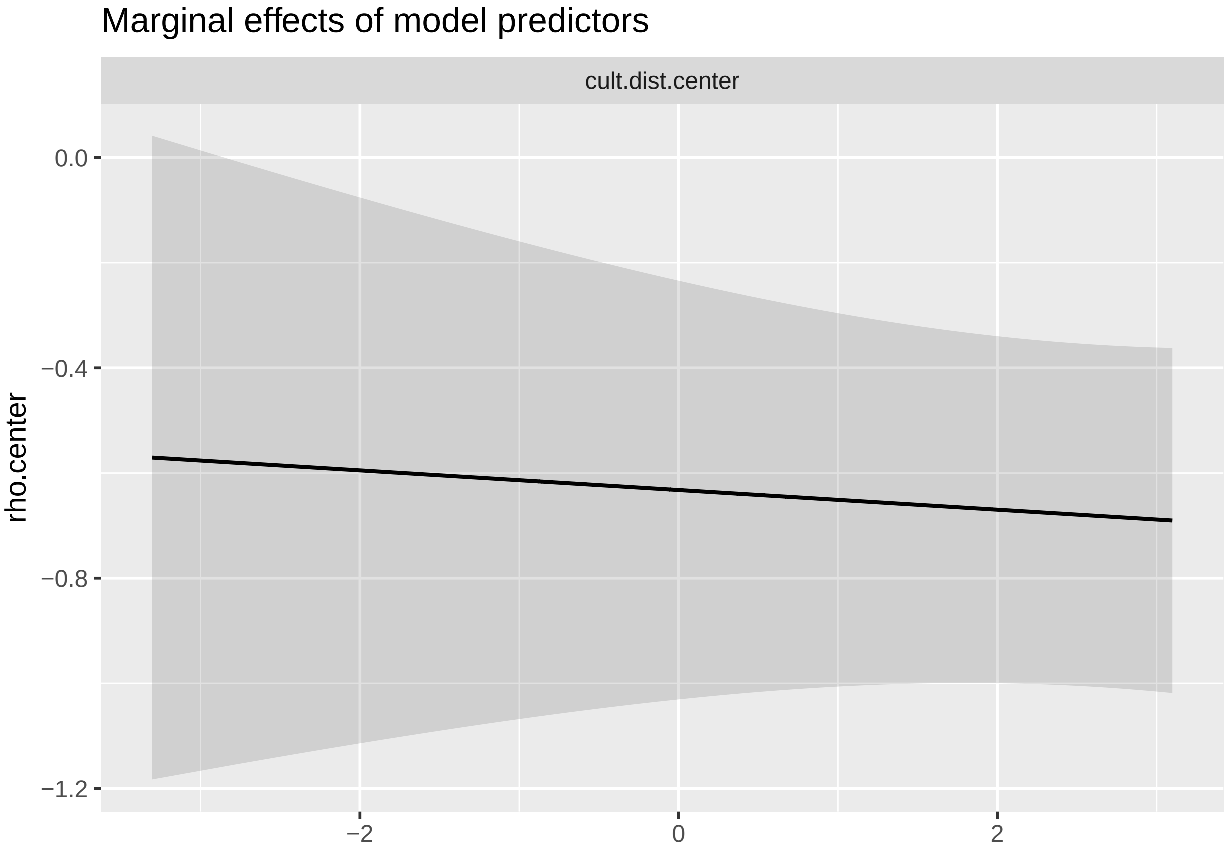
```
##
```

```
## $imputed_semantic_domain
```

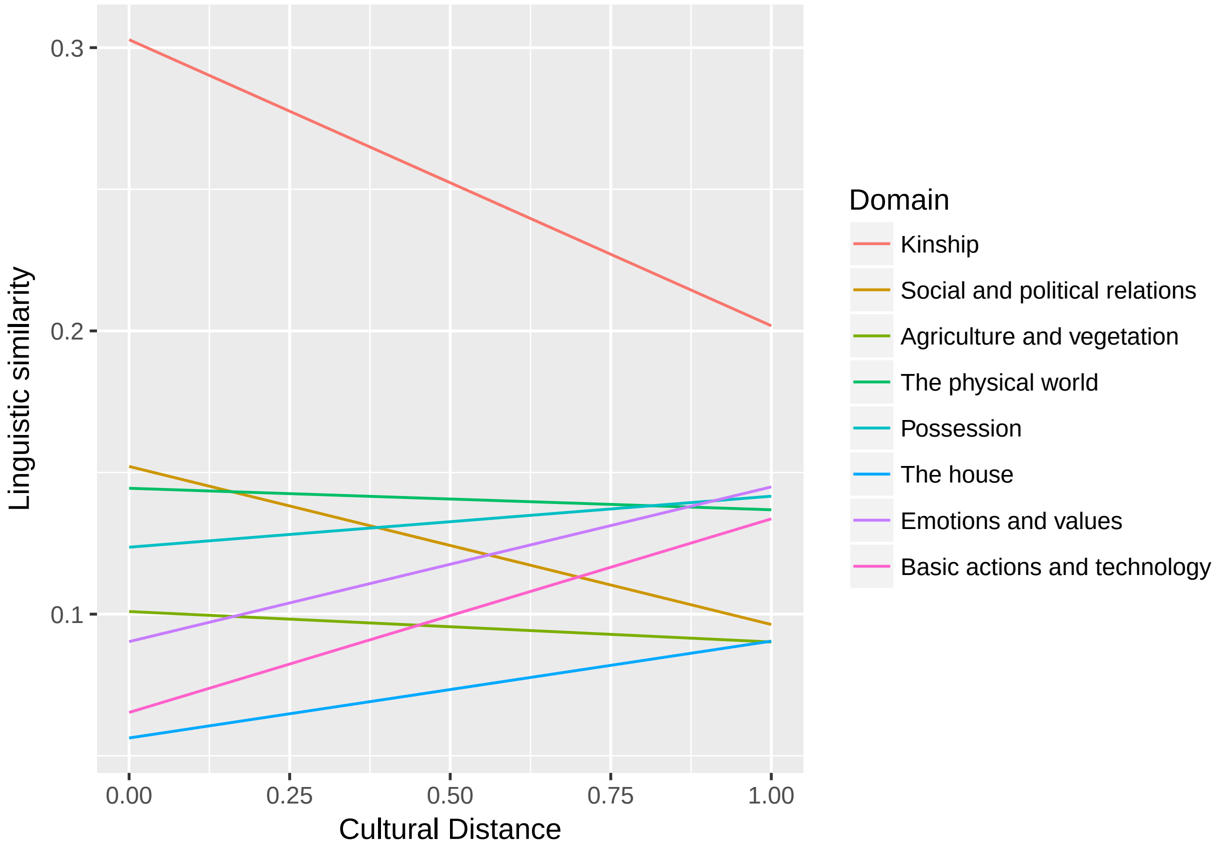


Plot the predicted relationships for each domain. The domains in the legend are sorted by the slope for cultural distance (greatest negative slope to greatest positive slope):

```
sjp.lmer(mD6,'eff', show.ci = T)
```



```
dom.order = ranef(mD6)$imputed_semantic_domain
dom.order = rownames(dom.order[order(dom.order$cult.dist.center),])
px = sjp.lmer(mD6,'rs.ri', show.ci = T, prnt.plot = F)
pdx = px$plot[[3]]$data
pdx$Domain = factor(pdx$grp, levels = dom.order)
pdx$x = pdx$x *
  attr(ling.dom$cult.dist.center,"scaled:scale") +
  attr(ling.dom$cult.dist.center,"scaled:center")
pdx$y = pdx$y *
  attr(ling.dom$rho.center,"scaled:scale") +
  attr(ling.dom$rho.center,"scaled:center")
ggplot(pdx,
       aes(x,y,colour=Domain)) +
  geom_line() +
  xlab("Cultural Distance") +
  ylab("Linguistic similarity")
```



Overall slope estimate for each domain:

```
ref.slopes.dom = data.frame(
  domain = rownames(ranef(mD6)$imputed_semantic_domain),
  slope = (fixef(mD6)["cult.dist.center"] +
  ranef(mD6)$imputed_semantic_domain[,2]))
ref.slopes.dom[order(ref.slopes.dom$slope),]
```

```
##                             domain        slope
## 4                          Kinship -0.163138433
## 6 Social and political relations -0.098486832
## 1     Agriculture and vegetation -0.034074828
## 8             The physical world -0.029551168
## 5                     Possession  0.007048736
## 7                      The house  0.030195874
## 3             Emotions and values  0.059418771
## 2   Basic actions and technology  0.079046059
```

Mantel tests

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

Data prep

Load historical 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 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)]
```

Convert the linguistic similarities to a matrix. This uses `igraph` to make an undirected graph from the long format with `local_alignment` as the edge weights, then output a matrix of adjacencies.

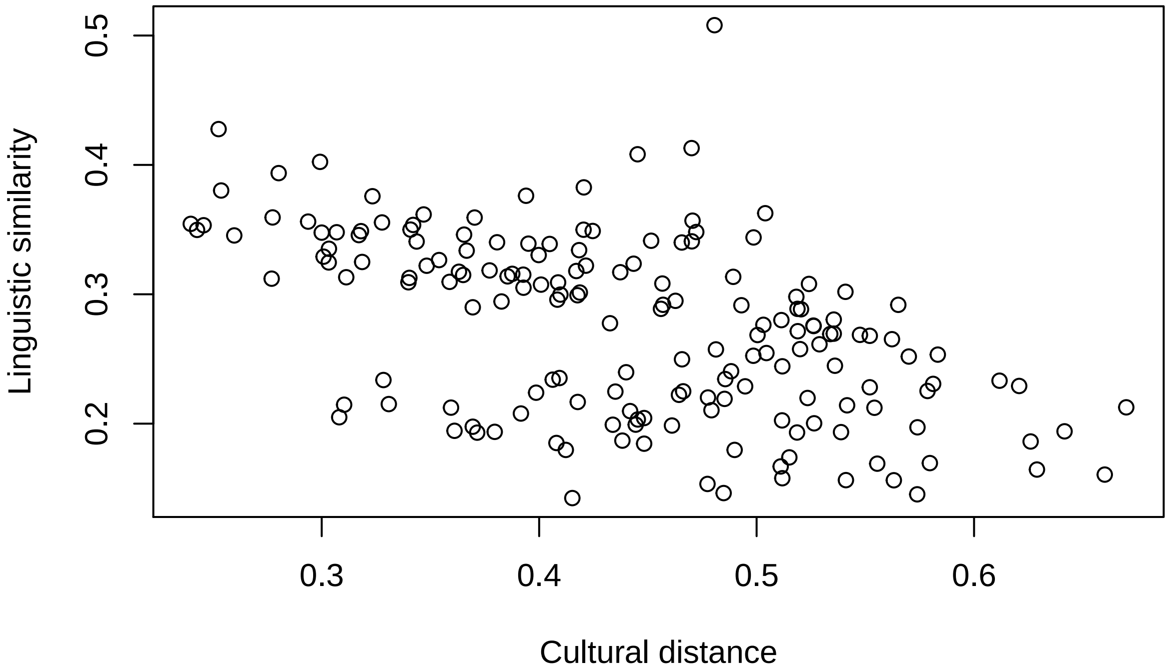
```
grph <- graph.data.frame(ling[,c("l1",'l2','local_alignment')], directed=FALSE)
# add value as a weight attribute
ling.m = get.adjacency(grph, attr="local_alignment", sparse=FALSE)
rownames(ling.m) = l[match(rownames(ling.m),l$iso2),]$Language2
colnames(ling.m) = l[match(colnames(ling.m),l$iso2),]$Language2
```

Match the distance matrices

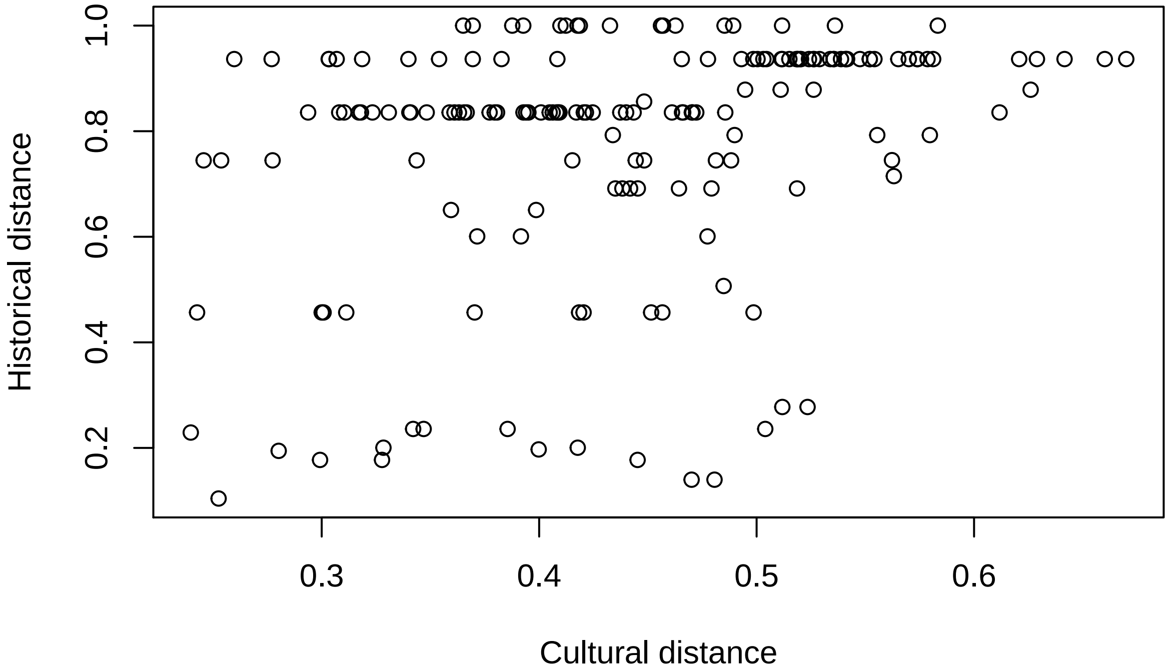
```
in.analysis = intersect(rownames(ling.m),rownames(cult.m))
in.analysis = intersect(in.analysis, rownames(hist.m))
cult.m2 = cult.m[in.analysis,in.analysis]
ling.m2 = ling.m[in.analysis,in.analysis]
hist.m2 = hist.m[in.analysis,in.analysis]
```

Note that there are only 19 languages with data on lingusitic, cultural and historical distance.

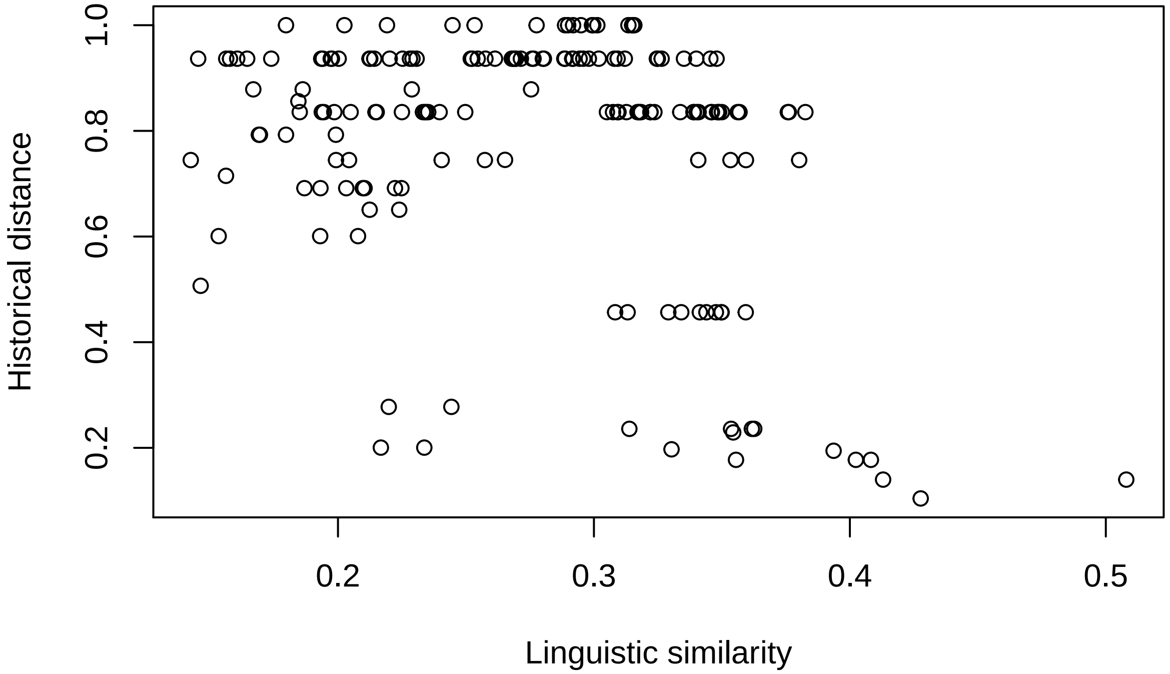
```
plot(as.dist(cult.m2),as.dist(ling.m2),
     xlab="Cultural distance",
     ylab="Linguistic similarity")
```



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



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



Tests

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

Correlation between cultural and linguistic distances:

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

```
##    mantelr      pval1      pval2      pval3  llim.2.5% ulim.97.5%
## -0.5243289  0.9948700  0.0051400  0.0051700 -0.6278127 -0.3821098
```

Correlation between cultural and historical distances:

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

```
##    mantelr      pval1      pval2      pval3  llim.2.5% ulim.97.5%
##  0.3243830  0.0128500  0.9871600  0.0137800  0.2286808  0.4280939
```

Correlation between linguistic and historical distances:

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

```
##    mantelr      pval1      pval2      pval3  llim.2.5% ulim.97.5%
## -0.3372882  0.9863800  0.0136300  0.0163000 -0.5054090 -0.1742784
```

Run a mantel test comparing the Linguistic similaritys to the cultural distances, controlling for the historical distance between languages:

```r
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  0.9896600  0.0103500  0.0111900 -0.6027438 -0.3108281
```

Run a mantel test comparing the linguistic similarities to the historical similarities, controlling for cultural distance:

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

```
##     mantelr       pval1       pval2       pval3   llim.2.5%  ulim.97.5%
## -0.20758646  0.92203000  0.07798000  0.14086000 -0.41440025  0.04182923
```

Are the relevant D-place features most predictive?

Long format to line up comparisons:

```
ling.dom.wide = ling.dom[,c("l1",'l2',
            'imputed_semantic_domain',
            "local_alignment","cult.dist")]
ling.dom.wide = reshape(ling.dom.wide,
                        idvar = c("l1","l2"),
                        timevar = "imputed_semantic_domain",
                        direction = "wide")
ling.dom.wide = cbind(ling.dom.wide[,1:2],
                      ling.dom.wide[,3:ncol(ling.dom.wide)][
                        order(names(ling.dom.wide[,3:ncol(ling.dom.wide)]))
                      ])
snames = c("Agri","Tech","Emot","Kin","Poss","Soc","Hous","Wrld")
names(ling.dom.wide) = c("l1","l2",
                         paste0("L.",snames),
                         paste0("C.",snames))
```

Raw correlation between each pair of domains

```
compareAllDomains =
  cor(ling.dom.wide[,
      grepl("L\\.",names(ling.dom.wide))],
    ling.dom.wide[,
      grepl("C\\.",names(ling.dom.wide))],
    use="complete.obs")

round(compareAllDomains,2)
```

```
##        C.Agri C.Tech C.Emot C.Kin C.Poss C.Soc C.Hous C.Wrld
## L.Agri  -0.47  -0.39  -0.42 -0.41  -0.33 -0.48  -0.34  -0.52
## L.Tech  -0.26  -0.23  -0.26 -0.34  -0.18 -0.27  -0.21  -0.33
## L.Emot  -0.27  -0.19  -0.27 -0.42  -0.18 -0.26  -0.23  -0.36
## L.Kin   -0.28  -0.18  -0.17 -0.27  -0.19 -0.27  -0.19  -0.28
## L.Poss  -0.44  -0.39  -0.42 -0.52  -0.34 -0.48  -0.38  -0.53
## L.Soc   -0.51  -0.35  -0.42 -0.51  -0.33 -0.49  -0.39  -0.53
## L.Hous  -0.30  -0.16  -0.14 -0.21  -0.18 -0.23  -0.18  -0.24
## L.Wrld  -0.18  -0.12  -0.08 -0.10  -0.09 -0.15  -0.04  -0.17
```

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.