Cultural distances: controlling for history

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

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

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("../data/FAIR/semantic_distances_FAIR_extended.csv", stringsAsFactors = F)
```

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

Each observation is now assocaited with a language family pair:

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

```
##    l1 l2       rho                family.group
## 1  ab os 0.5598228  Abkhaz-Adyge:Indo-European
## 3  ru uk 0.5304785 Indo-European:Indo-European
## 11 fr es 0.4220242 Indo-European:Indo-European
## 13 be uk 0.4155878 Indo-European:Indo-European
## 16 ba tt 0.4135547               Turkic:Turkic
## 19 be ru 0.4039364 Indo-European:Indo-European
```

And the same is true for area:

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

```
##      l1 l2           rho                              area.group
## 1463 cv ta  0.0676247885                         Indic:Inner Asia
## 1468 kl lv  0.0553355729               E North America:Inner Asia
## 1470 cv ko  0.0385081347                  Inner Asia:N Coast Asia
## 1476 ab he -0.0008438875 Greater Mesopotamia:Greater Mesopotamia
## 1479 kl ml -0.0234157394                   E North America:Indic
```

```
## 1484 kl is -0.1224497895                  E North America:Europe
```

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 1519.0 1553.7 -751.52   1503.0
## m1  9 1510.9 1549.8 -746.44   1492.9 10.17      1   0.001428 **
## ---
## 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: 1499.8
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -5.7742 -0.5309  0.0815  0.5625  4.8101
##
## Random effects:
```

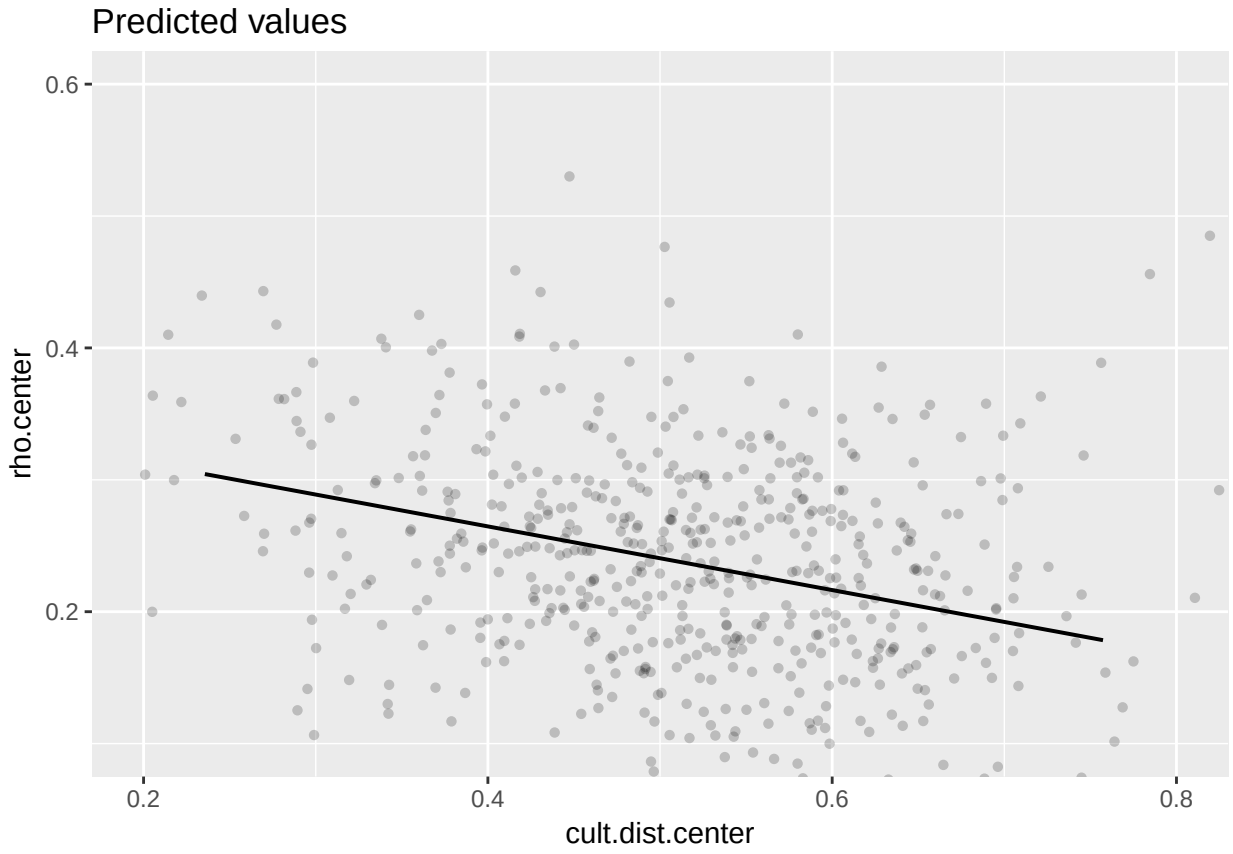
```
##  Groups       Name             Variance Std.Dev. Corr
##  family.group (Intercept)      0.03193  0.1787
##               cult.dist.center 0.03209  0.1791   -1.00
##  area.group   (Intercept)      0.03629  0.1905
##               cult.dist.center 0.04153  0.2038   -0.62
##  Residual                      0.79143  0.8896
## Number of obs: 561, groups:  family.group, 40; area.group, 20
##
## Fixed effects:
##                   Estimate Std. Error t value
## (Intercept)      -0.07791    0.08456  -0.921
## cult.dist.center -0.33923    0.08699  -3.900
##
## Correlation of Fixed Effects:
##             (Intr)
## clt.dst.cnt -0.626
```

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
```

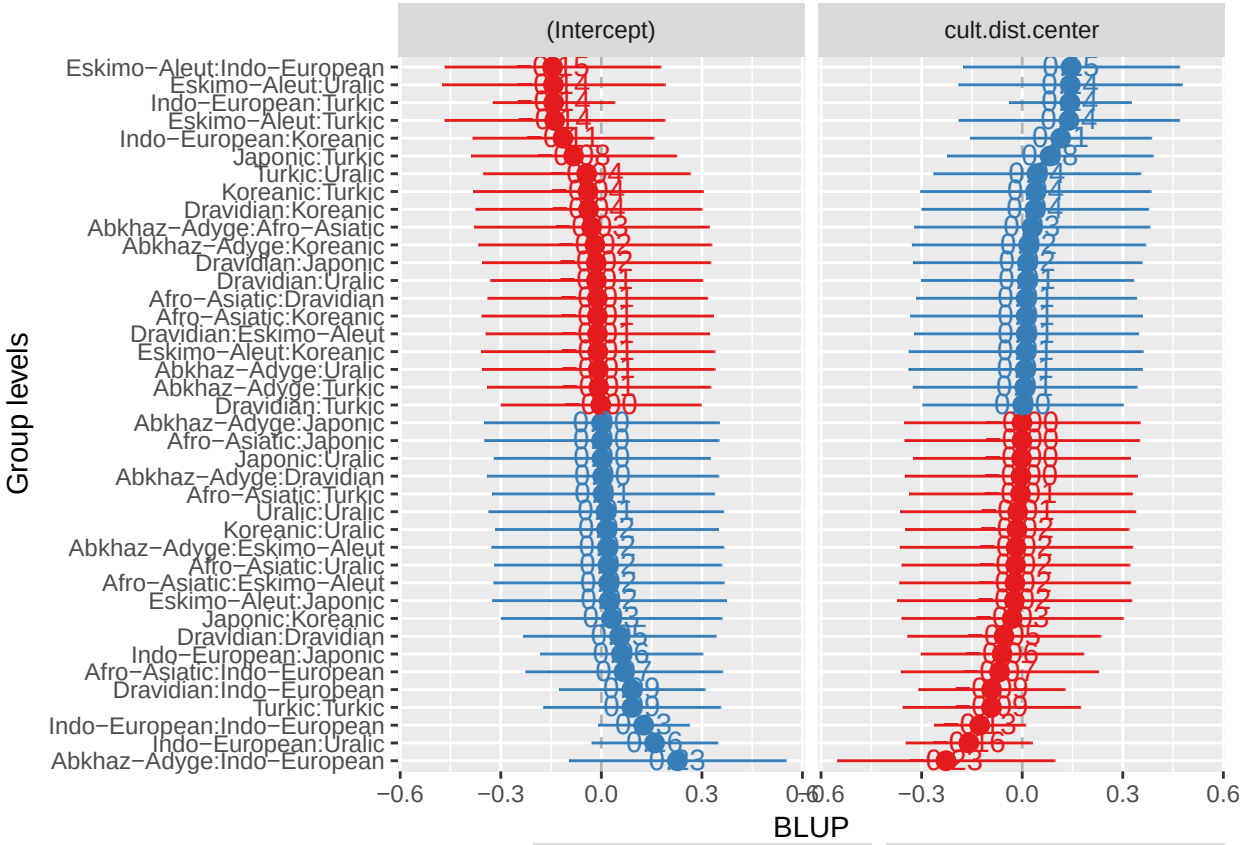
```
gx$plot$data$y = gx$plot$data$y *
  attr(ling$rho.center,"scaled:scale") +
  attr(ling$rho.center,"scaled:center")
gx$plot$data$resp.y = gx$plot$data$resp.y *
  attr(ling$rho.center,"scaled:scale") +
  attr(ling$rho.center,"scaled:center")
gx$plot$data$x = gx$plot$data$x *
  cdc.s +cdc.c
gx$plot + coord_cartesian(ylim=c(0.1,0.6),
                          xlim=c(0.2,0.8))
```

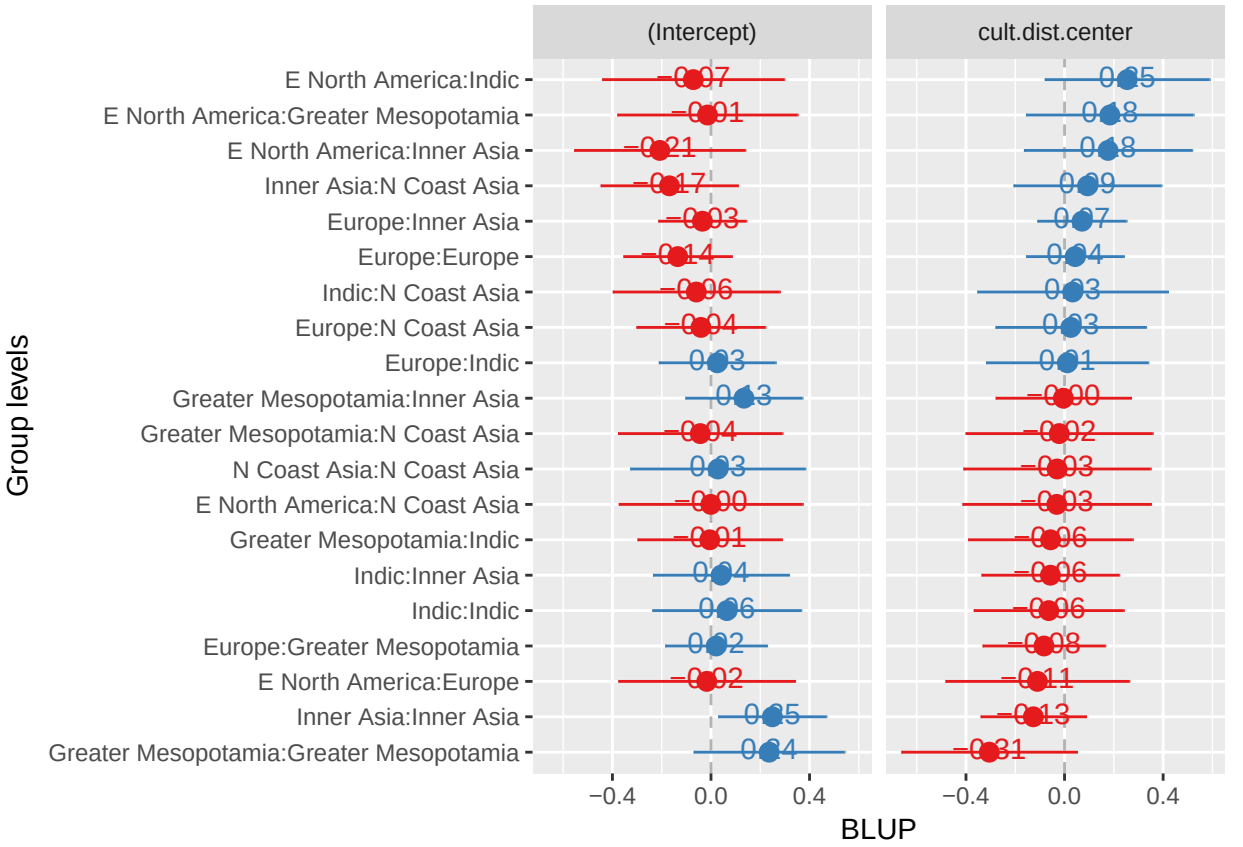


Plot the random effects:

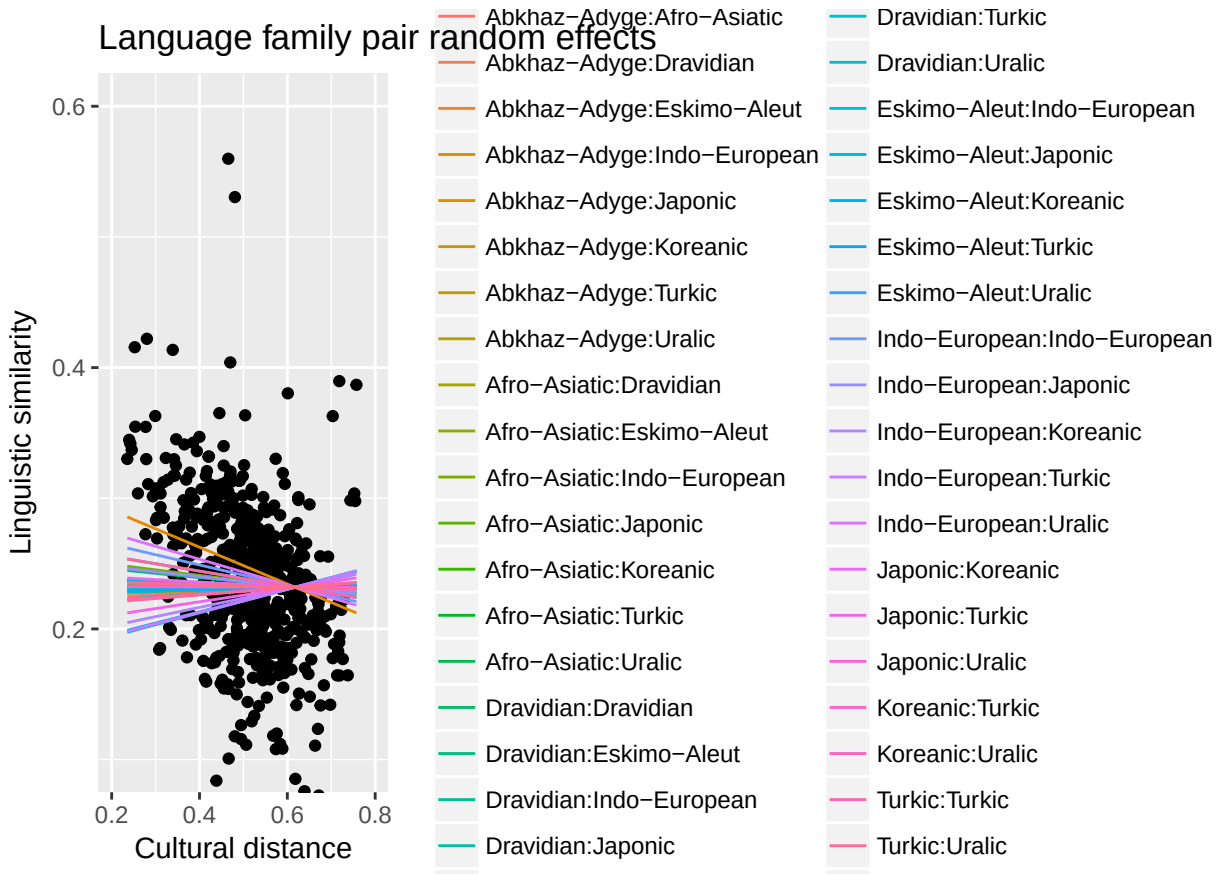
```
sjp.lmer(m1,'re', sort.est = "cult.dist.center")
```

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


(Intercept)
cult.dist.center
Group levels
Eskimo−Aleut:Indo−European
Eskimo−Aleut:Uralic
Indo−European:Turkic
Eskimo−Aleut:Turkic
Indo−European:Koreanic
Japonic:Turkic
Turkic:Uralic
Koreanic:Turkic
Dravidian:Koreanic
Abkhaz−Adyge:Afro−Asiatic
Abkhaz−Adyge:Koreanic
Dravidian:Japonic
Dravidian:Uralic
Afro−Asiatic:Dravidian
Afro−Asiatic:Koreanic
Dravidian:Eskimo−Aleut
Eskimo−Aleut:Koreanic
Abkhaz−Adyge:Uralic
Abkhaz−Adyge:Turkic
Dravidian:Turkic
Abkhaz−Adyge:Japonic
Afro−Asiatic:Japonic
Japonic:Uralic
Abkhaz−Adyge:Dravidian
Afro−Asiatic:Turkic
Uralic:Uralic
Koreanic:Uralic
Abkhaz−Adyge:Eskimo−Aleut
Afro−Asiatic:Uralic
Afro−Asiatic:Eskimo−Aleut
Eskimo−Aleut:Japonic
Japonic:Koreanic
Dravidian:Dravidian
Indo−European:Japonic
Afro−Asiatic:Indo−European
Dravidian:Indo−European
Turkic:Turkic
Indo−European:Indo−European
Indo−European:Uralic
Abkhaz−Adyge:Indo−European
−0.6
−0.3
0.0
0.3
0.6
BLUP

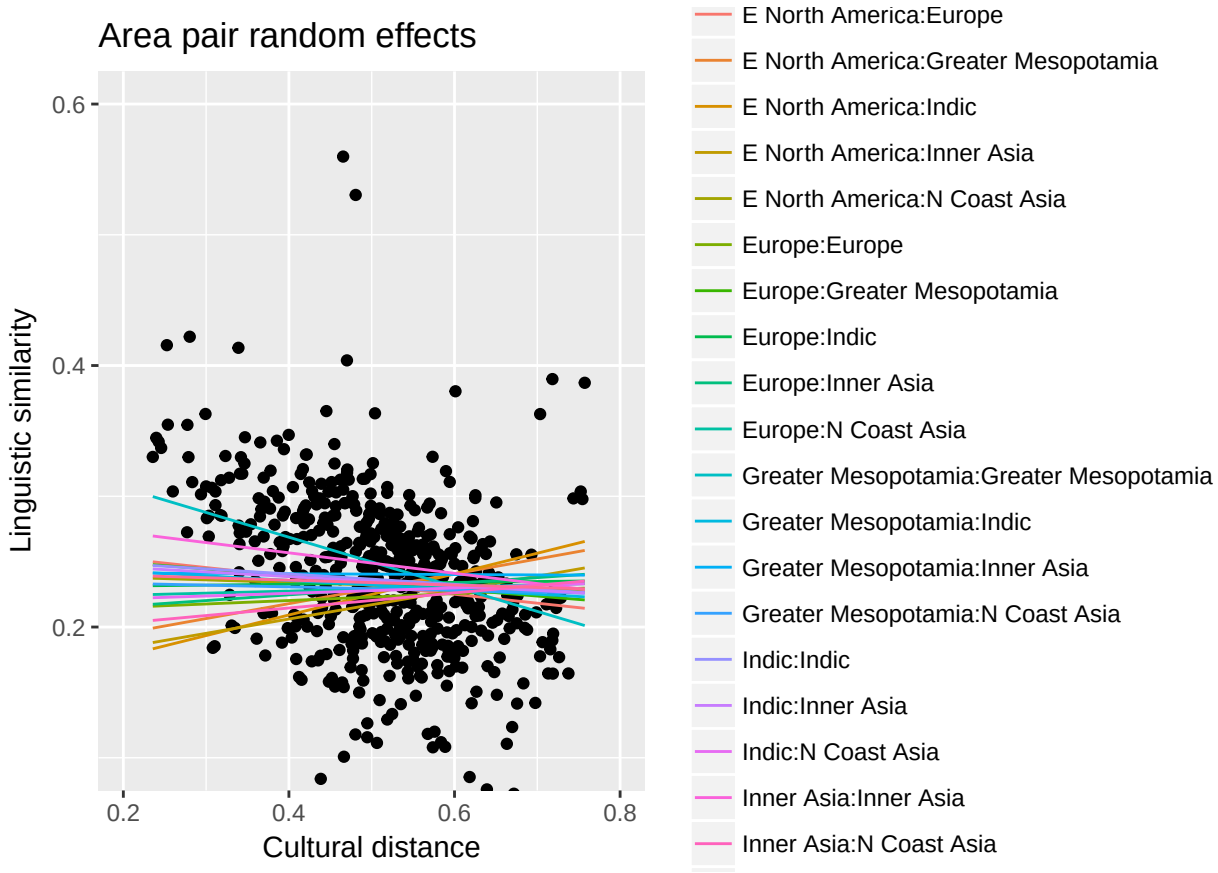

(Intercept)
cult.dist.center
Group levels
E North America:Indic
E North America:Greater Mesopotamia
E North America:Inner Asia
Inner Asia:N Coast Asia
Europe:Inner Asia
Europe:Europe
Indic:N Coast Asia
Europe:N Coast Asia
Europe:Indic
Greater Mesopotamia:Inner Asia
Greater Mesopotamia:N Coast Asia
N Coast Asia:N Coast Asia
E North America:N Coast Asia
Greater Mesopotamia:Indic
Indic:Inner Asia
Indic:Indic
Europe:Greater Mesopotamia
E North America:Europe
Inner Asia:Inner Asia
Greater Mesopotamia:Greater Mesopotamia
−0.4
0.0
0.4
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(rho))) +
  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))
```




```
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(rho))) +
```

```
  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 family are set to 0. We can check whether taking language family out makes a difference:

```
m0b = lmer(
  rho.center ~ 1 +
    (1 + cult.dist.center | area.group),
  data = ling
)
m1b = lmer(
  rho.center ~ 1 +
    cult.dist.center +
    (1 + cult.dist.center | area.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 | area.group)
## m1b: rho.center ~ 1 + cult.dist.center + (1 + cult.dist.center | area.group)
##     Df    AIC    BIC  logLik deviance  Chisq Chi Df Pr(>Chisq)
## m0b  5 1519.0 1540.7 -754.51   1509.0
## m1b  6 1508.7 1534.7 -748.35   1496.7 12.315      1  0.0004494 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(m1b)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula:
## rho.center ~ 1 + cult.dist.center + (1 + cult.dist.center | area.group)
##    Data: ling
##
## REML criterion at convergence: 1504.9
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -5.7067 -0.5631  0.0858  0.5612  4.9907
##
## Random effects:
##  Groups     Name             Variance Std.Dev. Corr
##  area.group (Intercept)      0.03792  0.1947
##             cult.dist.center 0.05925  0.2434   -0.69
##  Residual                    0.81161  0.9009
## Number of obs: 561, groups:  area.group, 20
##
## Fixed effects:
##                  Estimate Std. Error t value
## (Intercept)      -0.03863    0.07028   -0.55
## cult.dist.center -0.38523    0.08077   -4.77
##
## Correlation of Fixed Effects:
##             (Intr)
## clt.dst.cnt -0.511
```

The model is numerically almost exactly the same, so the p-value from the model comparison is just lower.

Tests within domains

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

```
ling.dom = read.csv(
  "../results/EA_distances/All_Domains_with_ling.csv",
  stringsAsFactors = F)

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

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

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 10311 10342 -5150.6    10301
## mD1  6 10293 10331 -5140.7    10281 19.9538      1  7.933e-06 ***
## mD2  7 10289 10333 -5137.7    10275  5.9424      1    0.01478 *
## ---
## 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 | 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)
## mD4: rho.center ~ 1 + (1 + cult.dist.center | family.group) + (1 |
## mD4:     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)
##     Df   AIC   BIC  logLik deviance  Chisq Chi Df Pr(>Chisq)
## mD2  7 10289 10333 -5137.7    10275
## mD4  7 10289 10333 -5137.7    10275 0.0000      0      1.000
## mD3  8 10291 10341 -5137.6    10275 0.1353      1      0.713
```

Random slopes for area do not improve the fit of the model.

Test random slopes for domain:

```
mdom1 = lmer(
  rho.center ~ 1 +
    (1 +cult.dist.center| family.group) +
    (1 | area.group) +
    (1 | imputed_semantic_domain),
  data = ling.dom)
mdom2 = lmer(
  rho.center ~ 1 +
    (1 +cult.dist.center| family.group) +
    (1 | 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:     area.group) + (1 | imputed_semantic_domain)
## mdom2: rho.center ~ 1 + (1 + cult.dist.center | family.group) + (1 |
## mdom2:     area.group) + (1 + cult.dist.center | imputed_semantic_domain)
##       Df   AIC   BIC  logLik deviance  Chisq Chi Df Pr(>Chisq)
## mdom1  7 10289 10333 -5137.7    10275
## mdom2  9 10280 10336 -5130.9    10262 13.619      2   0.001103 **
## ---
## 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:

```
mD5 = lmer(
  rho.center ~ 1 +
    (1 + cult.dist.center | family.group) +
    (1 | 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:     area.group) + (1 + cult.dist.center | imputed_semantic_domain)
## mD6: rho.center ~ (1 + cult.dist.center | family.group) + (1 | area.group) +
## mD6:     (1 + cult.dist.center | imputed_semantic_domain) + cult.dist.center
##     Df   AIC   BIC  logLik deviance  Chisq Chi Df Pr(>Chisq)
## mD5  9 10280 10336 -5130.9    10262
## mD6 10 10281 10343 -5130.6    10261 0.6717      1     0.4124
```

Summary of the final model, with random effects plot:

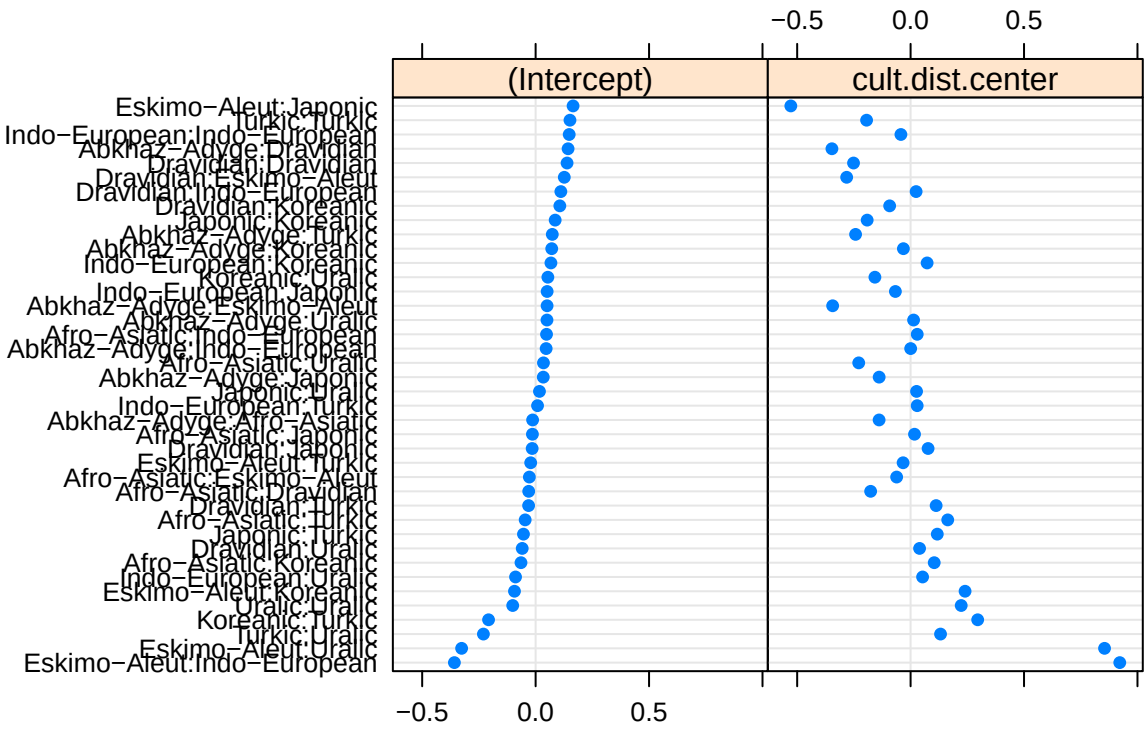
```
summary(mD6)
```

```
## Linear mixed model fit by REML ['lmerMod']
```

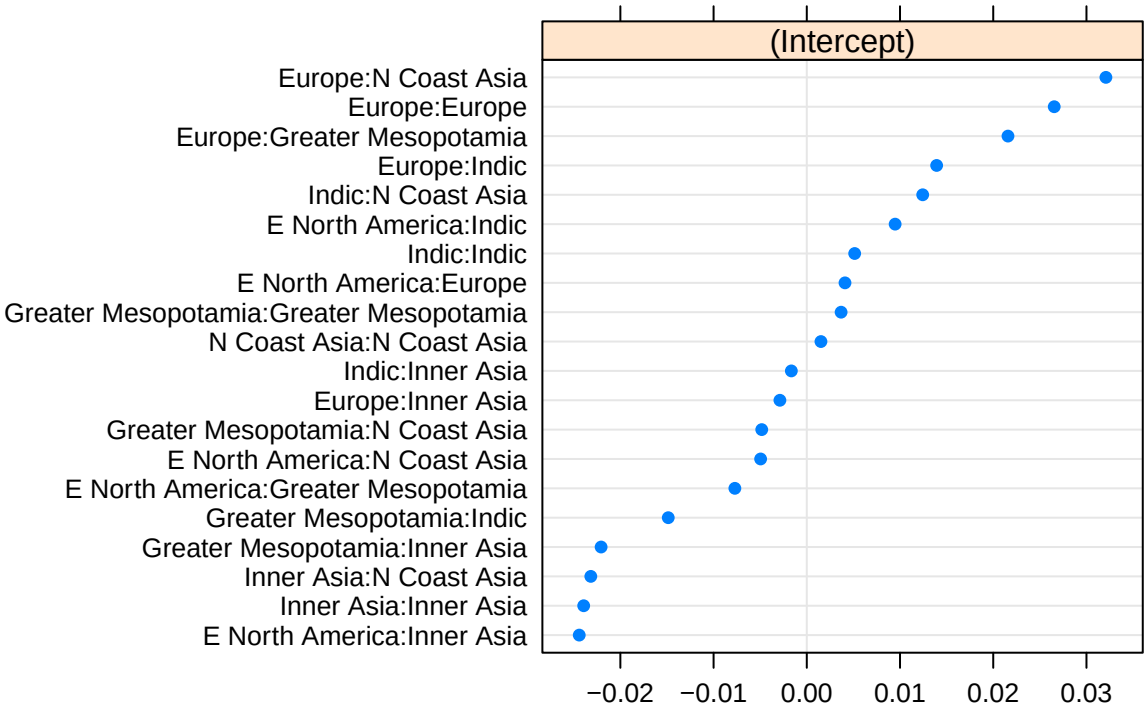
```
## Formula:
## rho.center ~ (1 + cult.dist.center | family.group) + (1 | area.group) +
##     (1 + cult.dist.center | imputed_semantic_domain) + cult.dist.center
##    Data: ling.dom
##
## REML criterion at convergence: 10268
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -5.2667 -0.3978  0.0508  0.4604  3.6029
##
## Random effects:
##  Groups                  Name             Variance Std.Dev. Corr
##  family.group            (Intercept)      0.031677 0.17798
##                          cult.dist.center 0.122253 0.34965  -0.76
##  area.group              (Intercept)      0.001677 0.04095
##  imputed_semantic_domain (Intercept)      0.027232 0.16502
##                          cult.dist.center 0.006224 0.07889  -0.33
##  Residual                                 0.930120 0.96443
## Number of obs: 3673, groups:
## family.group, 40; area.group, 20; imputed_semantic_domain, 7
##
## Fixed effects:
##                  Estimate Std. Error t value
## (Intercept)      -0.05603    0.07714  -0.726
## cult.dist.center -0.06561    0.07764  -0.845
##
## Correlation of Fixed Effects:
##             (Intr)
## clt.dst.cnt -0.438
```

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

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

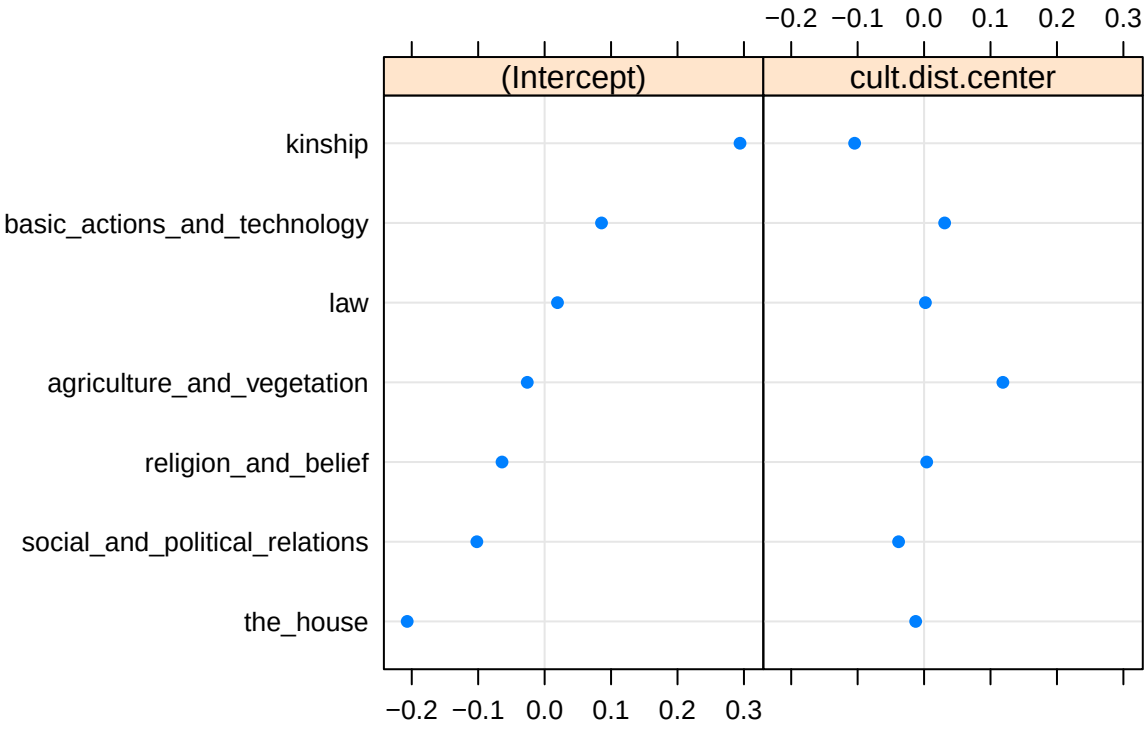

family.group
(Intercept)
cult.dist.center
−0.5
0.0
0.5
Eskimo−Aleut:Japonic
Turkic:Turkic
Indo−European:Indo−European
Abkhaz−Adyge:Dravidian
Dravidian:Dravidian
Dravidian:Eskimo−Aleut
Dravidian:Indo−European
Dravidian:Koreanic
Japonic:Koreanic
Abkhaz−Adyge:Turkic
Abkhaz−Adyge:Koreanic
Indo−European:Koreanic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

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


area.group
(Intercept)
Europe:N Coast Asia
Europe:Europe
Europe:Greater Mesopotamia
Europe:Indic
Indic:N Coast Asia
E North America:Indic
Indic:Indic
E North America:Europe
Greater Mesopotamia:Greater Mesopotamia
N Coast Asia:N Coast Asia
Indic:Inner Asia
Europe:Inner Asia
Greater Mesopotamia:N Coast Asia
E North America:N Coast Asia
E North America:Greater Mesopotamia
Greater Mesopotamia:Indic
Greater Mesopotamia:Inner Asia
Inner Asia:N Coast Asia
Inner Asia:Inner Asia
E North America:Inner Asia
−0.02
−0.01
0.00
0.01
0.02
0.03

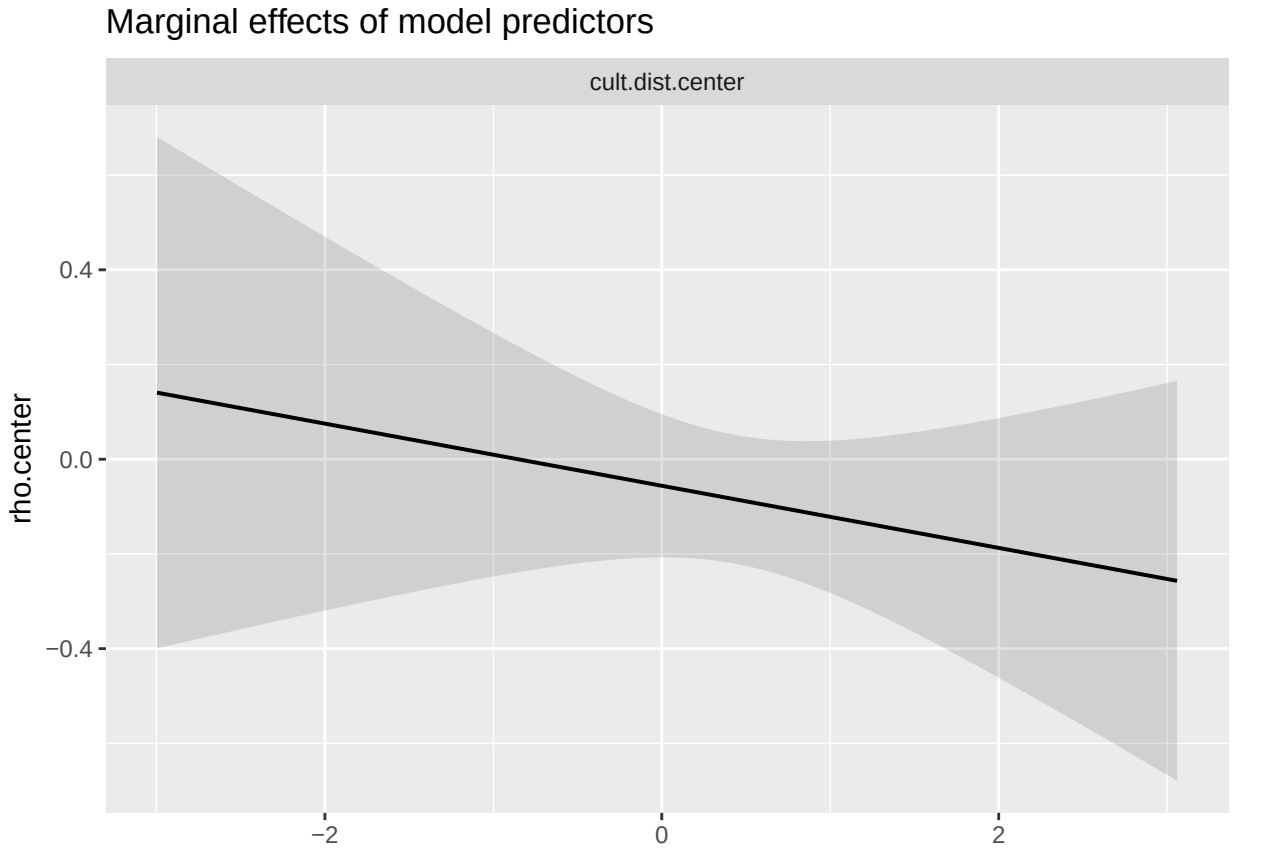
```
##
```

```
## $imputed_semantic_domain
```

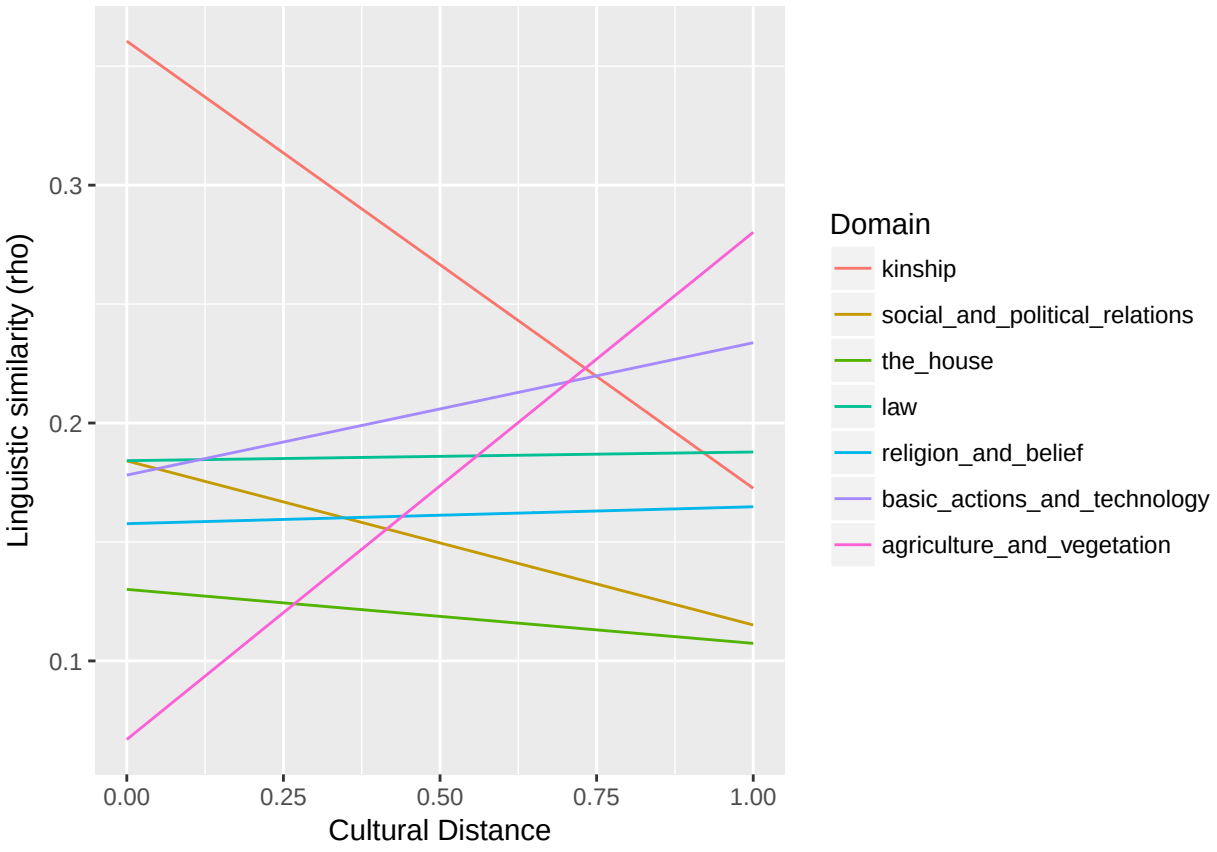


Plot the predicted relationships for each domain. The domains in the legend are sorted by the slope for cultural distance (greatest negative slope to greatest positive slope):

```
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$plot[[2]]$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 (rho)")
```



"Agriculture and vegetation" seems to be working differently from "Kinship".

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 `rho` as the edge weights, then output a matrix of adjacencies.

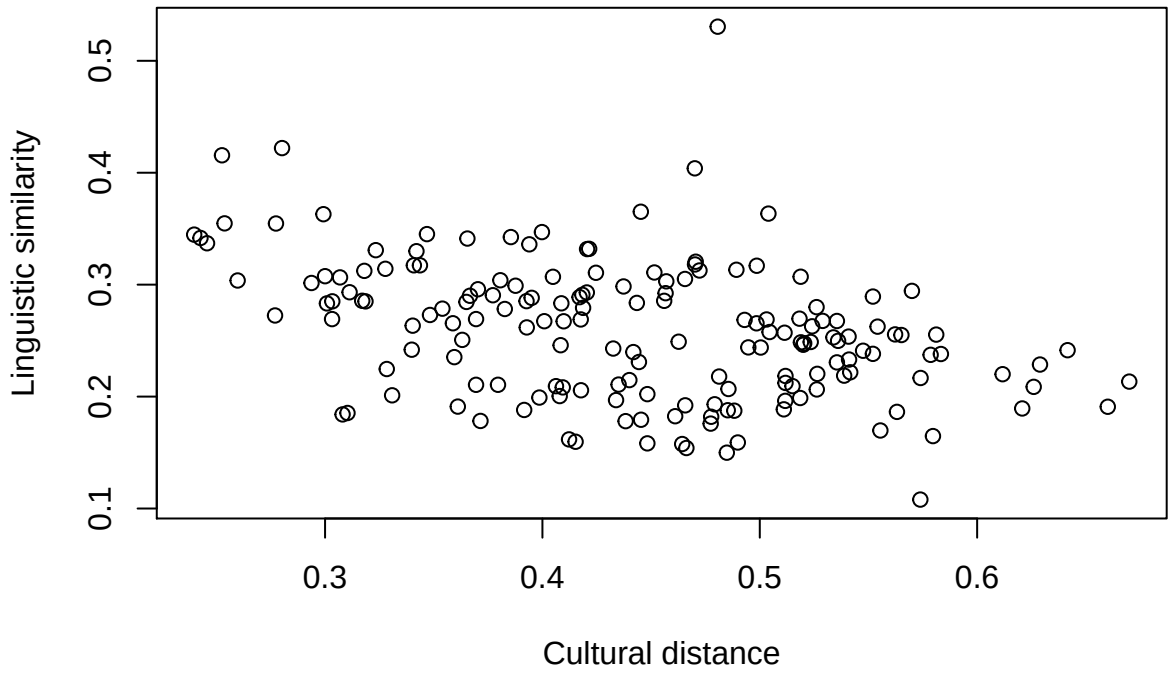
```
grph <- graph.data.frame(ling[,c("l1",'l2','rho')], directed=FALSE)
# add value as a weight attribute
ling.m = get.adjacency(grph, attr="rho", sparse=FALSE)
rownames(ling.m) = l[match(rownames(ling.m),l$iso2),]$Language2
colnames(ling.m) = l[match(colnames(ling.m),l$iso2),]$Language2
```

Match the distance matrices

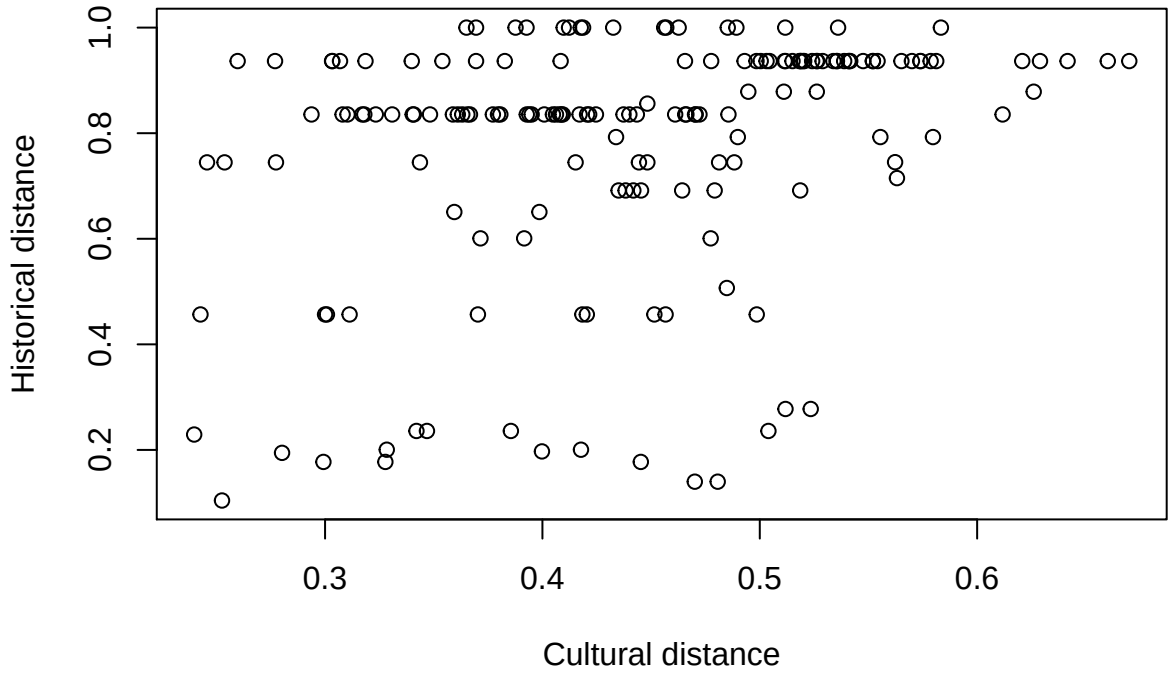
```
in.analysis = intersect(rownames(ling.m),rownames(cult.m))
in.analysis = intersect(in.analysis, rownames(hist.m))
cult.m2 = cult.m[in.analysis,in.analysis]
ling.m2 = ling.m[in.analysis,in.analysis]
hist.m2 = hist.m[in.analysis,in.analysis]
```

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

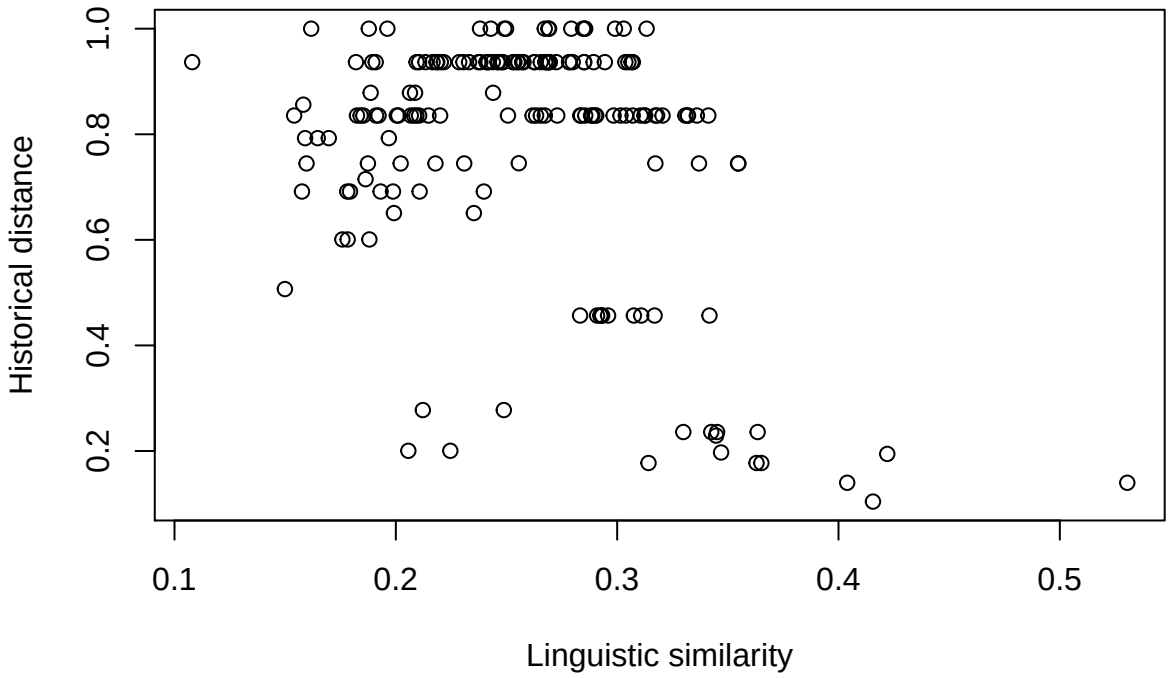
```
plot(as.dist(cult.m2),as.dist(ling.m2),
     xlab="Cultural distance",
     ylab="Linguistic similarity")
```




```
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

Simple correlation without control for history:

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

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

```
##    mantelr      pval1      pval2      pval3  llim.2.5% ulim.97.5%
## -0.4034202  0.9827300  0.0172800  0.0198600 -0.5437260 -0.3072486
```

Run a mantel test comparing the Linguistic similaritys to the cultural distances, 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%
## -0.3176622  0.9575600  0.0424500  0.0656300 -0.4582571 -0.2055098
```

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.