Cultural distances: Common crawl data

Contents

Introduction

This file replicates the tests for the main wikipedia data on the common crawl data.

Load libraries

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

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

```
datasetName = "cc"
lingDistancesFile = "../data/FAIR/nel-k100-cc-alignments-by-language-pair.csv"
lingDistancesFileNK = "../data/FAIR/nel-k100-cc-alignments-by-language-pair-without-kinsip.csv"
lingDistancesByDomainFile = "../results/EA_distances/nel-k100-cc_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 [fr, es, el, de, hu], so we'll remove these:

```
ling = ling[!(ling$l1 %in% c("fr","es","el","de","hu")),]
ling = ling[!(ling$l2 %in% c("fr","es","el","de","hu")),]
```

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

```r
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:

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

```
##      l1  l2 local_alignment          family.group
## 3   myv  cv      0.05112061          Turkic:Uralic
## 5    la  cv      0.06646598 Indo-European:Turkic
## 6    cv sah      0.06823760          Turkic:Turkic
## 13   ga  cv      0.07949409 Indo-European:Turkic
## 19   cv  he      0.08650789   Afro-Asiatic:Turkic
## 20   cv  te      0.08761480       Dravidian:Turkic
```

And the same is true for area:

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

```
##      l1 l2 local_alignment             area.group
## 1112 uk ja       0.3804179 Inner Asia:N Coast Asia
## 1113 be ru       0.3821532   Inner Asia:Inner Asia
## 1115 uk be       0.4022741   Inner Asia:Inner Asia
## 1116 cs uk       0.4376378       Europe:Inner Asia
## 1118 cs ru       0.4581089       Europe:Inner Asia
## 1119 uk ru       0.5460480   Inner Asia:Inner Asia
```

Number of observations:

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

```
## [1] 302
```

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

```
## [1] 31
```

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

```
## [1] 7
```

```r
# 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
##    Afro-Asiatic      0                   1     0          0            0
##    Dravidian         0                   0     3          0            0
##    Indo-European     8                   2     1          5            0
##    Japonic           0                   0     0          0            1
##    Sino-Tibetan      0                   0     0          0            0
##    Turkic            0                   1     0          5            0
##    Uralic            0                   0     0          3            0
##
##                 Southeast Asia
##    Afro-Asiatic               0
##    Dravidian                  0
##    Indo-European              0
##    Japonic                    0
##    Sino-Tibetan               1
##    Turkic                     0
##    Uralic                     0
```

LMER models

Mixed effects model, predicting Linguistic similarity 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
)
m0.5 = lmer(
  rho.center ~ 1 +
    comparison_count.center +
    (1 + cult.dist.center | family.group) +
    (1 + cult.dist.center | area.group),
  data = ling
)
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
)
anova(m0,m0.5,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)
## 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     8 802.77 832.46 -393.39   786.77
## m0.5   9 784.04 817.43 -383.02   766.04 20.737      1  5.269e-06 ***
## m1    10 782.16 819.27 -381.08   762.16  3.871      1    0.04913 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Cultural similarity is significantly correlated with Linguistic similarity. 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: 771.9
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.8836 -0.8267  0.0172  0.7516  4.3332
##
## Random effects:
##  Groups        Name             Variance Std.Dev. Corr
##  family.group (Intercept)      0.258996 0.50892
##               cult.dist.center 0.025975 0.16117  1.00
##  area.group   (Intercept)      0.060083 0.24512
##               cult.dist.center 0.009889 0.09944  1.00
##  Residual                      0.647454 0.80465
## Number of obs: 302, groups:  family.group, 23; area.group, 19
##
## Fixed effects:
##                        Estimate Std. Error t value
## (Intercept)             0.01581    0.15367   0.103
## comparison_count.center 0.28774    0.05771   4.986
## cult.dist.center        0.16254    0.07824   2.077
##
## Correlation of Fixed Effects:
##             (Intr) cmpr_.
## cmprsn_cnt.  0.097
## clt.dst.cnt  0.665 -0.043
```

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 similarity") +
  ylab("Semantic alignment") +
  ggtitle("")
gx
```
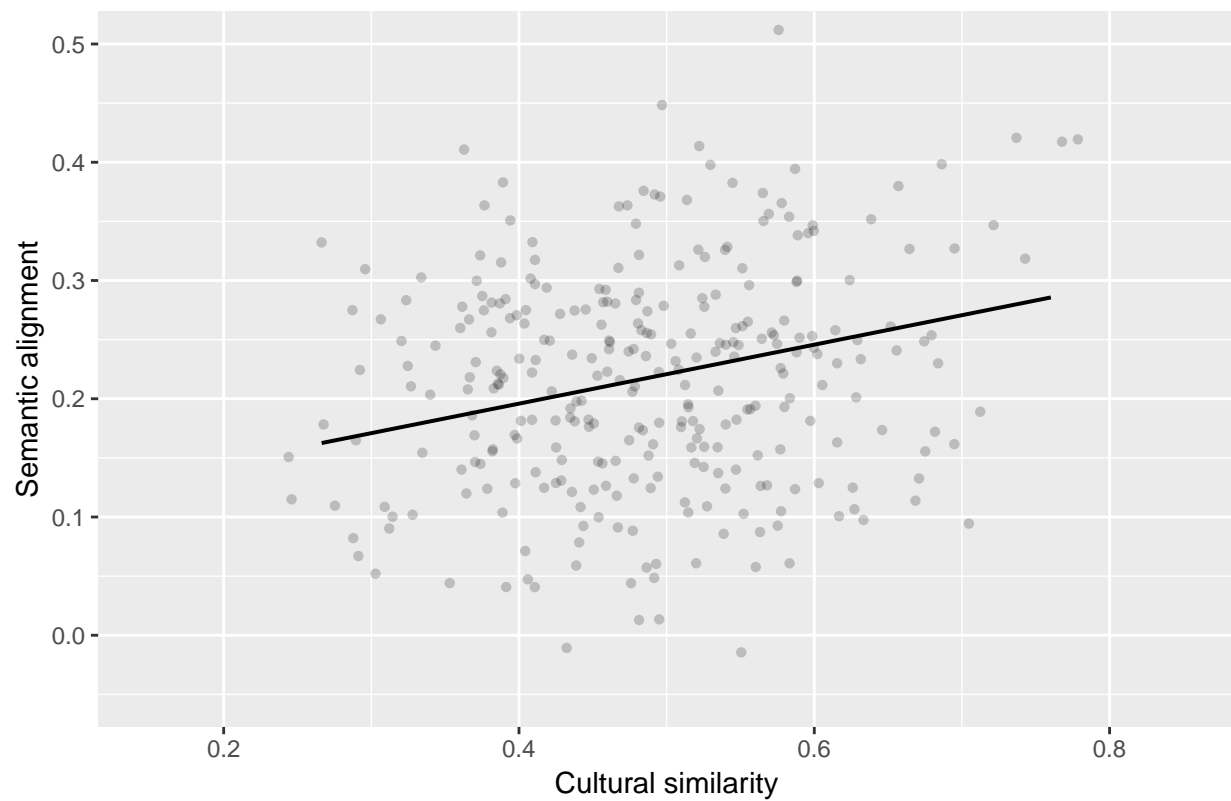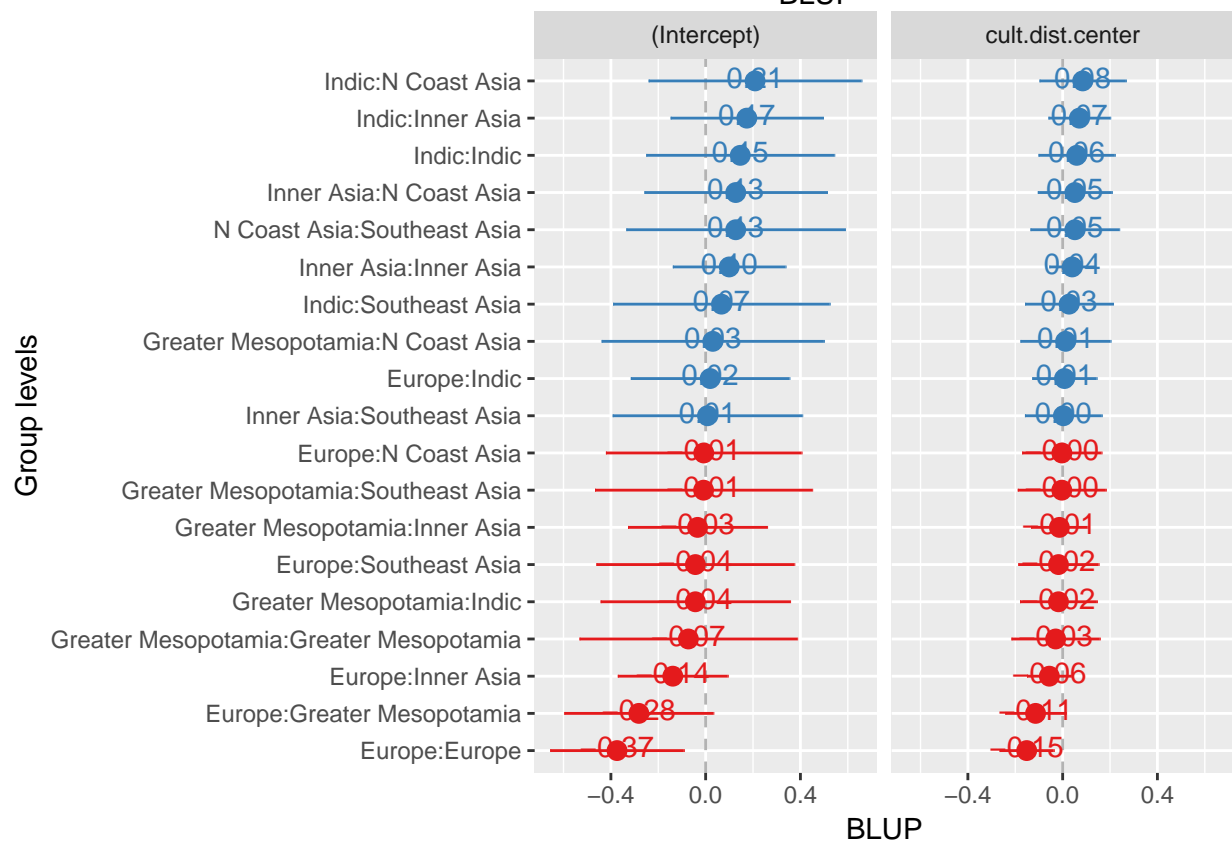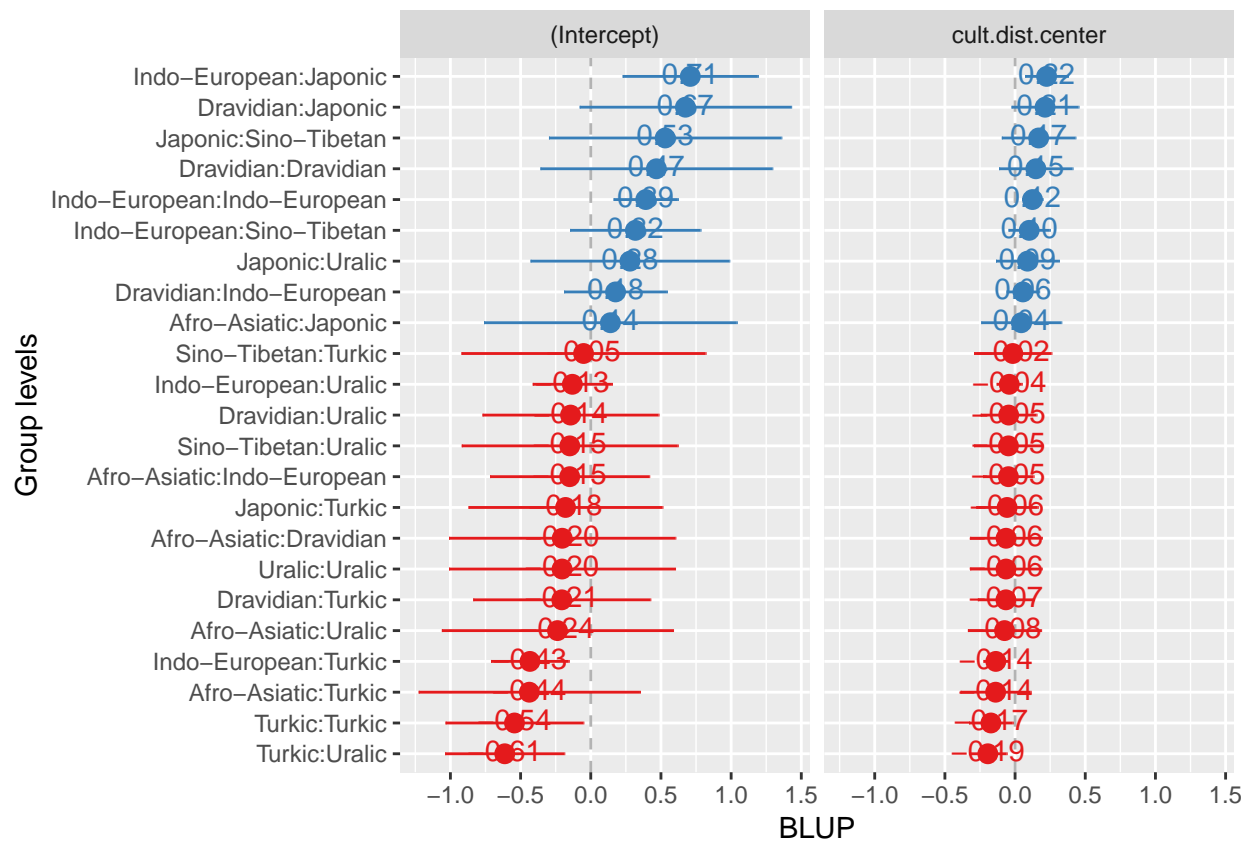
```
pdf(paste0("../results/stats/",datasetName,"/CulturalDistance_Rho_Graph.pdf"),
    height=2.5, width=2.5)
gx
dev.off()
```

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

Plot the random effects:

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

```
## Plotting random effects...
## Plotting random effects...
```
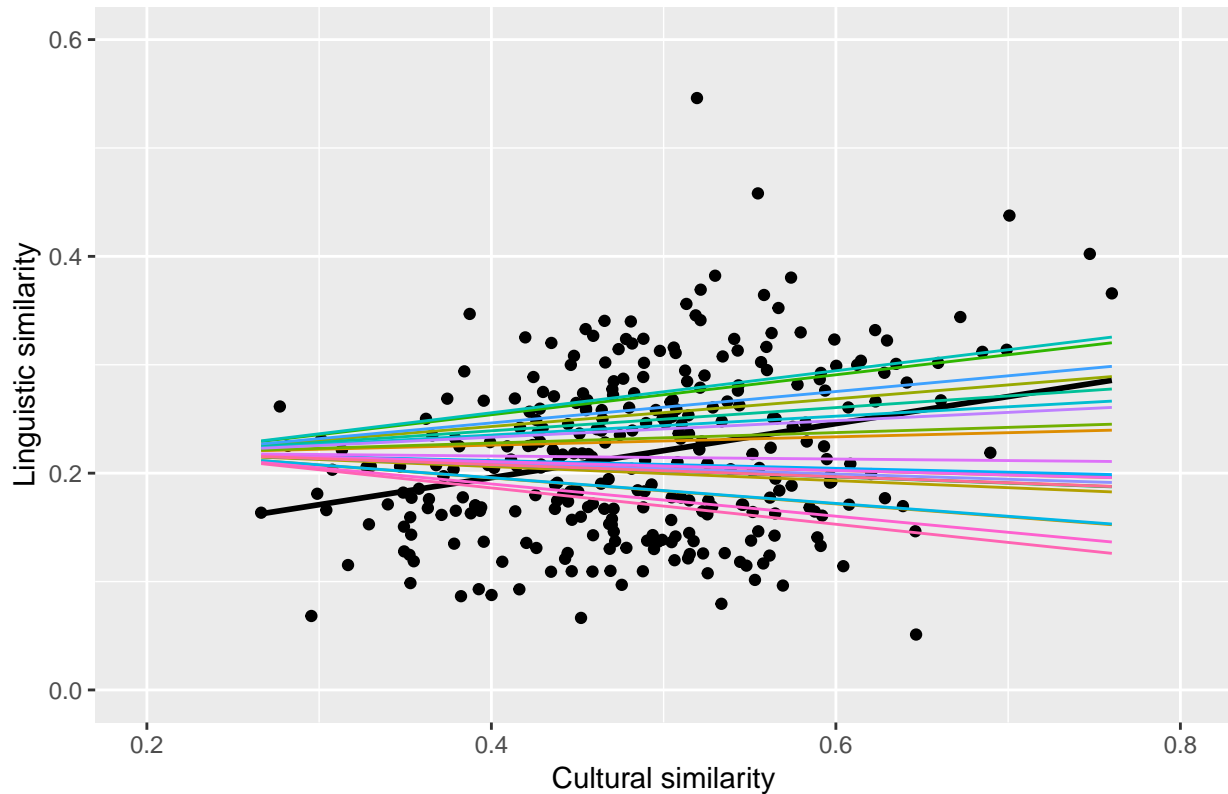
```
px = sjp.lmer(m1,'rs.ri', prnt.plot = F)
dx = px$plot[[1]]$data
dx$x = dx$x * cdc.s + cdc.c
dx$y = dx$y *
  attr(ling$rho.center,"scaled:scale") +
  attr(ling$rho.center,"scaled:center")

ggplot(dx,aes(x,y)) +
  geom_point(data=ling,
   mapping=aes(x=as.numeric(cult.dist),
              y=as.numeric(local_alignment))) +
  stat_smooth(data=gx$data,method="lm",colour="black",
             se=F)+
  geom_line(mapping = aes(colour=grp)) +
  xlab("Cultural similarity")+
  ylab("Linguistic similarity") +
  ggtitle("Language family pair random effects") +
  coord_cartesian(ylim=c(0.0,0.6),
                 xlim=c(0.2,0.8)) +
  theme(legend.position = "none")
```

## Language family pair random effects


```
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))) +
  stat_smooth(data=gx$data,method="lm",colour="black",
              se=F)+
  geom_line(mapping = aes(colour=grp)) +
  xlab("Cultural similarity")+
  ylab("Linguistic similarity") +
  ggtitle("Area pair random effects") +
  coord_cartesian(ylim=c(0.1,0.6),
                  xlim=c(0.2,0.8))
```
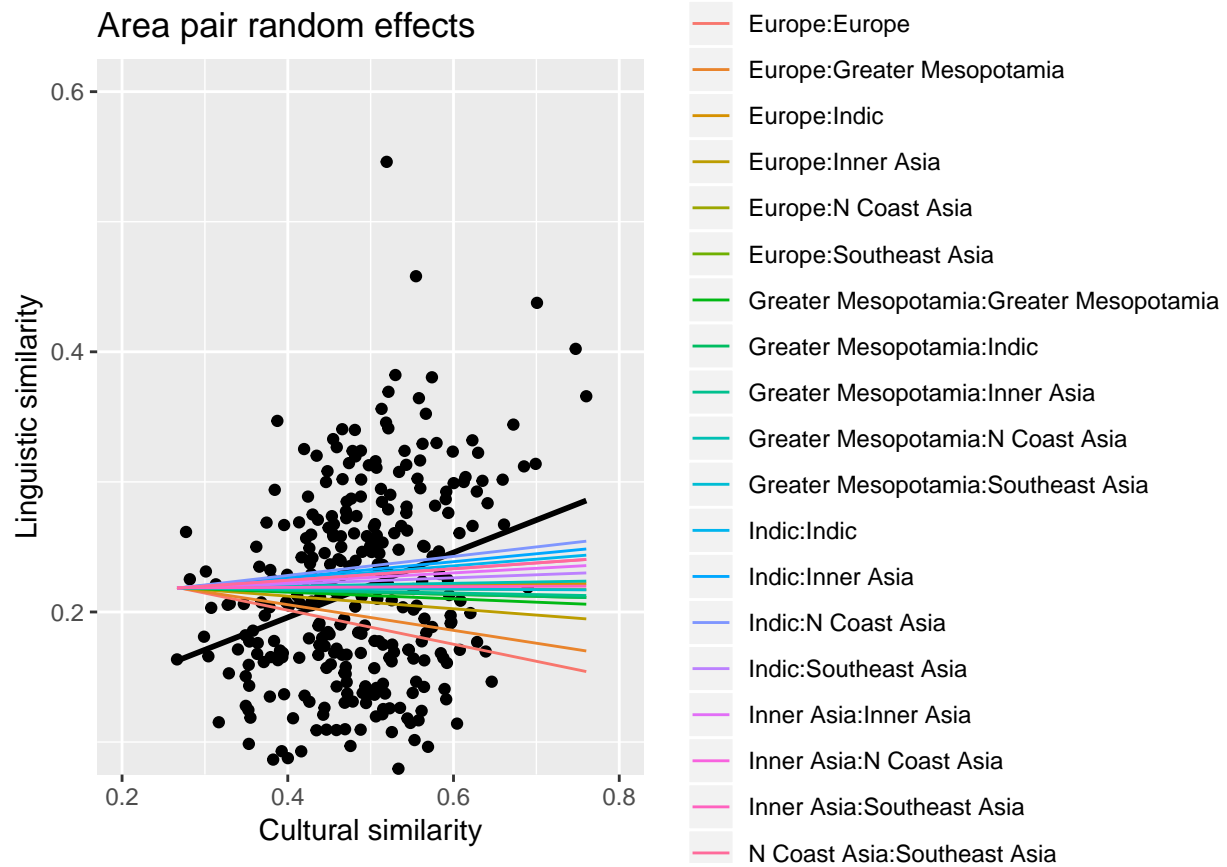
Without Kinship data

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:

```
mONK = lmer(
  rho.center ~ 1 +
    (1 + cult.dist.center | family.group) +
    (1 + cult.dist.center | area.group),
  data = lingNK
)
m0.5NK = lmer(
  rho.center ~ 1 +
    comparison_count.center +
    (1 + cult.dist.center | family.group) +
    (1 + cult.dist.center | area.group),
  data = lingNK
)
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
)
anova(mONK,m0.5NK,m1NK)
```

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

## Data: lingNK
## Models:
## mONK: rho.center ~ 1 + (1 + cult.dist.center | family.group) + (1 +
## mONK:     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)
## mONK     8 802.77 832.46 -393.39   786.77
## m0.5NK   9 784.04 817.43 -383.02   766.04 20.737      1  5.269e-06 ***
## m1NK    10 782.16 819.27 -381.08   762.16  3.871      1    0.04913 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
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: 771.9
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.8836 -0.8267  0.0172  0.7516  4.3332
##
## Random effects:
##  Groups        Name             Variance Std.Dev. Corr
##  family.group (Intercept)      0.258996 0.50892
##               cult.dist.center 0.025975 0.16117  1.00
##  area.group   (Intercept)      0.060083 0.24512
##               cult.dist.center 0.009889 0.09944  1.00
##  Residual                      0.647454 0.80465
## Number of obs: 302, groups:  family.group, 23; area.group, 19
##
## Fixed effects:
##                        Estimate Std. Error t value
## (Intercept)             0.01581    0.15367   0.103
## comparison_count.center 0.28774    0.05771   4.986
## cult.dist.center        0.16254    0.07824   2.077
##
## Correlation of Fixed Effects:
##            (Intr) cmpr_.
## cmprsn_cnt.  0.097
## clt.dst.cnt  0.665 -0.043
```

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 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 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
```

Load the geographic distances:

```
geoDist = read.csv("../data/GeographicDistances.csv",stringsAsFactors = F)
geoDist.m = as.matrix(geoDist)
# Convert to log distance
geoDist.m = log(geoDist.m)
geoDist.m[is.infinite(geoDist.m)] = 0
rownames(geoDist.m) = colnames(geoDist.m)
```

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 16 languages with data on lingusitic, cultural and historical distance.
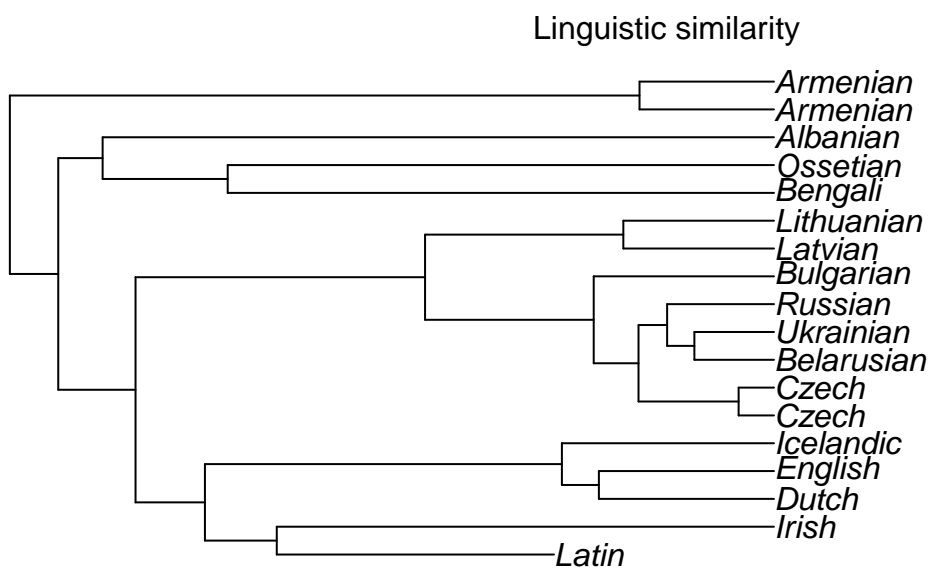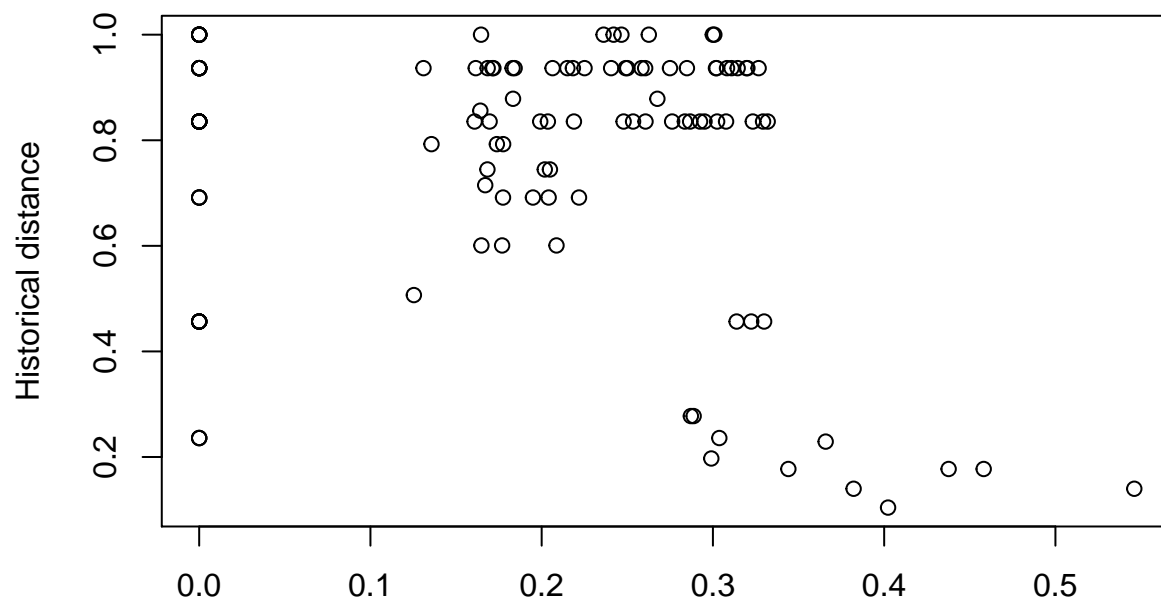
```
plot(as.dist(cult.m2),as.dist(ling.m2),
     xlab="Cultural similarity",
     ylab="Linguistic similarity")
```
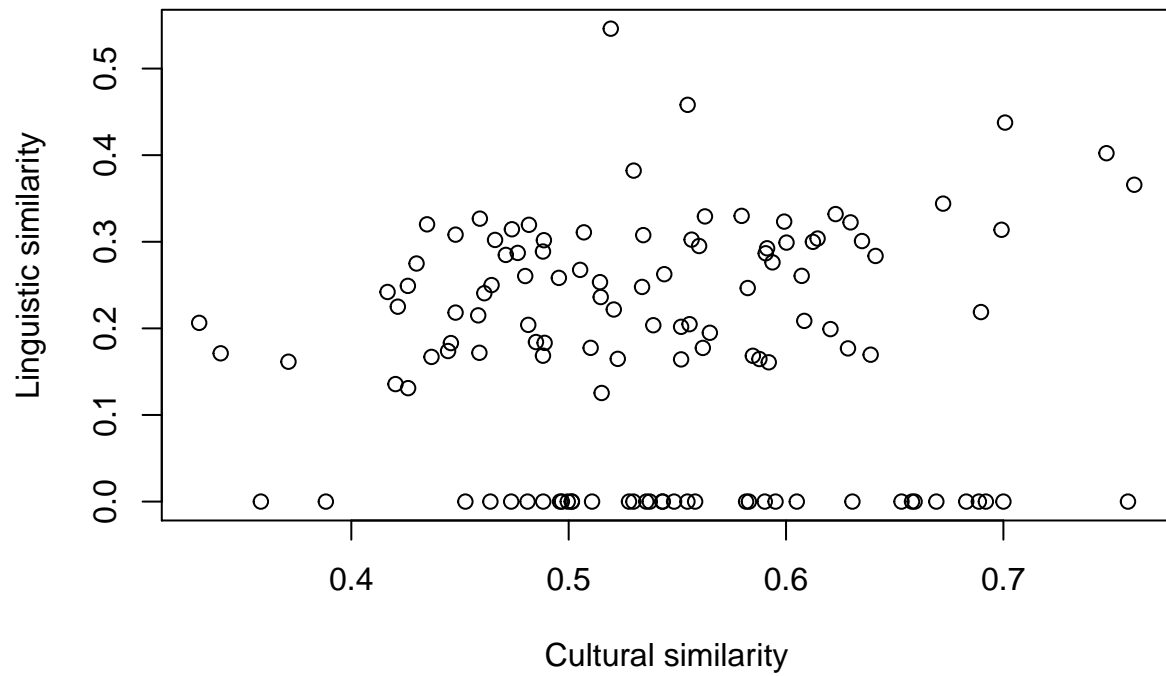
```
plot(as.dist(cult.m2),as.dist(hist.m2),
     xlab="Cultural similarity",
     ylab="Historical distance")
```
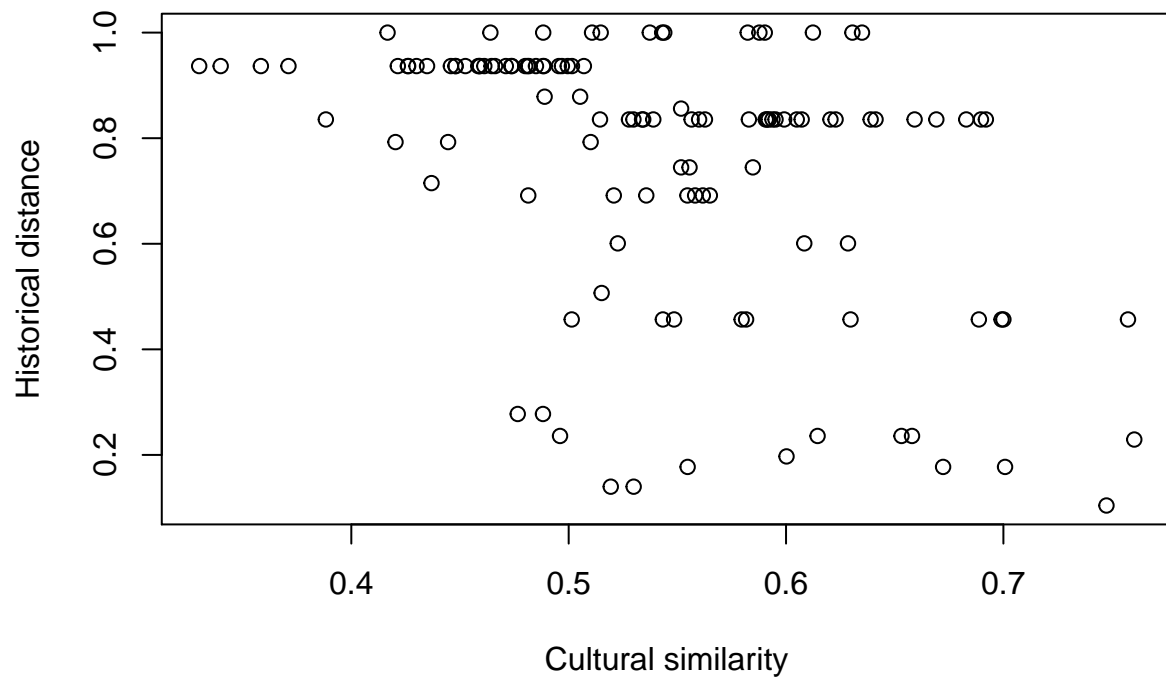


```
plot(as.dist(ling.m2),as.dist(hist.m2),
     xlab="Linguistic similarity",
     ylab="Historical distance")
```

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)
```

```
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){
    print(paste("Correlation between",
                names(distms)[i],"and",names(distms)[j]))
    print(ecodist::mantel(as.dist(distms[[i]]) ~
              as.dist(distms[[j]]),
              nperm = 100000))
  }
}
```

```
## [1] "Correlation between Cultrual and Linguistic"
##      mantelr       pval1        pval2        pval3     llim.2.5%
##  0.007089021   0.476090000   0.523920000   0.952700000 -0.116606375
##    ulim.97.5%
##  0.114570879
## [1] "Correlation between Cultrual and Historical"
##    mantelr       pval1      pval2       pval3  llim.2.5% ulim.97.5%
## -0.4506997   0.9959000   0.0041100   0.0041100 -0.5576863 -0.3312250
## [1] "Correlation between Cultrual and Geographic"
##    mantelr       pval1      pval2       pval3  llim.2.5% ulim.97.5%
## -0.4862387   0.9969900   0.0030200   0.0030200 -0.5830547 -0.3598362
## [1] "Correlation between Linguistic and Historical"
##      mantelr       pval1       pval2        pval3   llim.2.5%   ulim.97.5%
## -0.19469321   0.96706000   0.03295000   0.06612000 -0.34802529 -0.07862642
## [1] "Correlation between Linguistic and Geographic"
##      mantelr       pval1       pval2        pval3   llim.2.5%   ulim.97.5%
##  0.14129069   0.10016000   0.89985000   0.20482000 -0.01111971   0.25902135
## [1] "Correlation between Historical and Geographic"
##    mantelr       pval1      pval2       pval3  llim.2.5% ulim.97.5%
##  0.4771207   0.0006100   0.9994000   0.0006100  0.3548048  0.5787921
```

Run a mantel test comparing the Linguistic 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%
```

```
## -0.09211939  0.78746000  0.21255000  0.42539000 -0.18659805 -0.00754341
```

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

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

```
##      mantelr        pval1        pval2        pval3     llim.2.5%
##  0.001986124  0.487440000  0.512570000  0.986720000 -0.078389477
##    ulim.97.5%
##  0.095838595
```

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 linguistic similarity in a given domain to the overall cultural distance. The linguistic similarity 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.

The full results are in the file:

`../results/stats/cc/Cor_LingAlignmentByDomains_vs_OverallCulturalSimilarity.csv`

Summary:

`p1res`

```
##                                    Domain       Beta          p Adjusted p sig
## 16                              The body 0.15134409 0.004149489 0.08713927
## 2                              The house 0.17975342 0.008594577 0.18048613
## 4     Miscellaneous function words 0.25622891 0.009549536 0.20054026
## 18                         Food and drink 0.13190747 0.010172759 0.21362794
## 7                            Modern world 0.16181687 0.013670009 0.28707018
## 10                       Sense perception 0.13895889 0.014530466 0.30513978
## 1                              Possession 0.14011895 0.016813474 0.35308295
## 12                       Spatial relations 0.18350843 0.023159309 0.48634548
## 20                                 Animals 0.16700875 0.028940003 0.60774007
## 11                                    Time 0.20926434 0.048324561 1.00000000
## 14 Social and political relations 0.12553287 0.050196868 1.00000000
## 9                   Clothing and grooming 0.13879079 0.060482835 1.00000000
## 6                     Speech and language 0.11474155 0.070469675 1.00000000
## 8                               Cognition 0.10952235 0.077245162 1.00000000
## 3         Agriculture and vegetation 0.07684041 0.150595089 1.00000000
## 15     Basic actions and technology 0.06812223 0.159678882 1.00000000
## 21                                Quantity 0.19094189 0.179160048 1.00000000
## 17                       The physical world 0.07468933 0.184553956 1.00000000
## 13                     Emotions and values 0.06384113 0.246747491 1.00000000
## 19                                 Kinship 0.09856782 0.292750110 1.00000000
## 5                                   Motion 0.02736510 0.573863349 1.00000000
```

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/cc/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
## 156     Animals    Politics 0.3250611 0.01116537
```
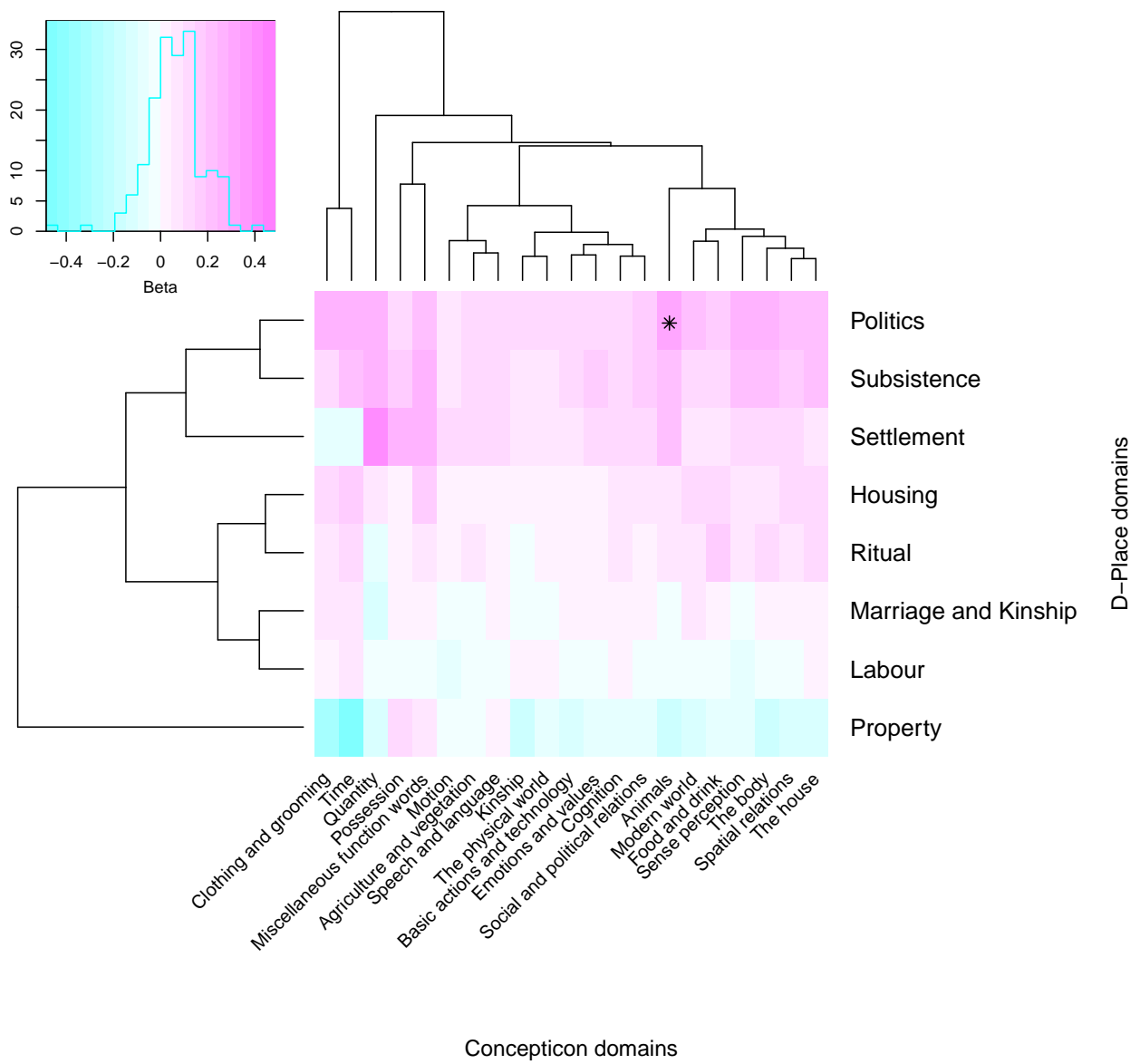
Figure 1:

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

This test compares each linguistic similarity scores 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.
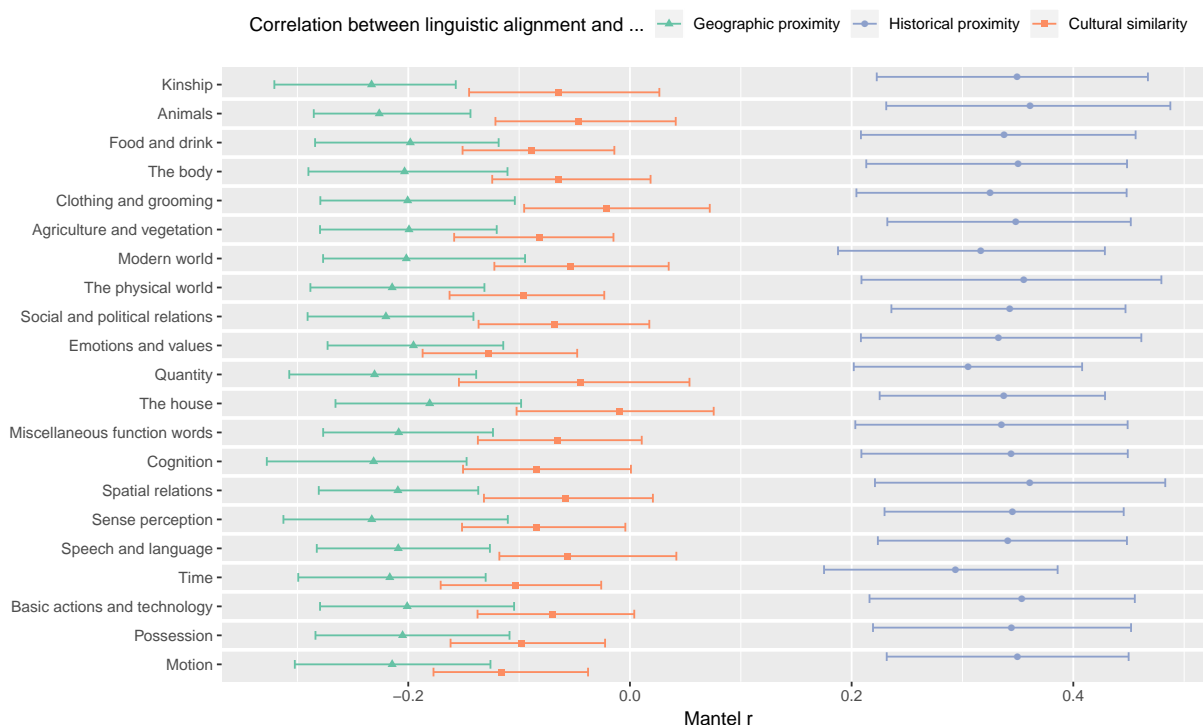

Figure 2:

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.