

Estimating the typological distribution of word order and sentence particles

Load libraries.

```
library(lme4)
```

```
## Warning: package 'lme4' was built under R version 3.2.5
```

```
## Loading required package: Matrix
```

```
## Warning: package 'Matrix' was built under R version 3.2.5
```

Load data.

```
d.wals = read.csv("wals_data/language.csv", na.string='')
```

Make a variable which marks the position of the verb.

```
OrderSOV = as.character(d.wals$X81A.Order.of.Subject..Object.and.Verb)
```

```
d.wals$VerbPos = NA
d.wals[OrderSOV %in% c('1 SOV', '6 OSV'),]$VerbPos = 3
d.wals[OrderSOV %in% c('2 SVO', '5 OVS'),]$VerbPos = 2
d.wals[OrderSOV %in% c('3 VSO', '4 VOS'),]$VerbPos = 1
```

Make a centered variable from the verb position.

```
d.wals$VerbPos.norm = d.wals$VerbPos/3
```

Make a variable for the position of particles. Code only initial, final and none, removing 'second position' and 'other position'.

```
# make a shortcut for position of polar question particles
QP = as.character(d.wals$X92A.Position.of.Polar.Question.Particles)
```

```
# find non-NA datapoints
QPNA = !is.na(d.wals$X92A.Position.of.Polar.Question.Particles)
```

```
d.wals$QP.pos3 = NA
d.wals$QP.pos3[QPNA & QP=='1 Initial'] = "3 Initial"
d.wals$QP.pos3[QPNA & QP=='3 Second position'] = NA
d.wals$QP.pos3[QPNA & QP=='4 Other position'] = NA
d.wals$QP.pos3[QPNA & QP=='2 Final'] = "2 Final"
d.wals$QP.pos3[QP=='6 No question particle'] = "1 None"
```

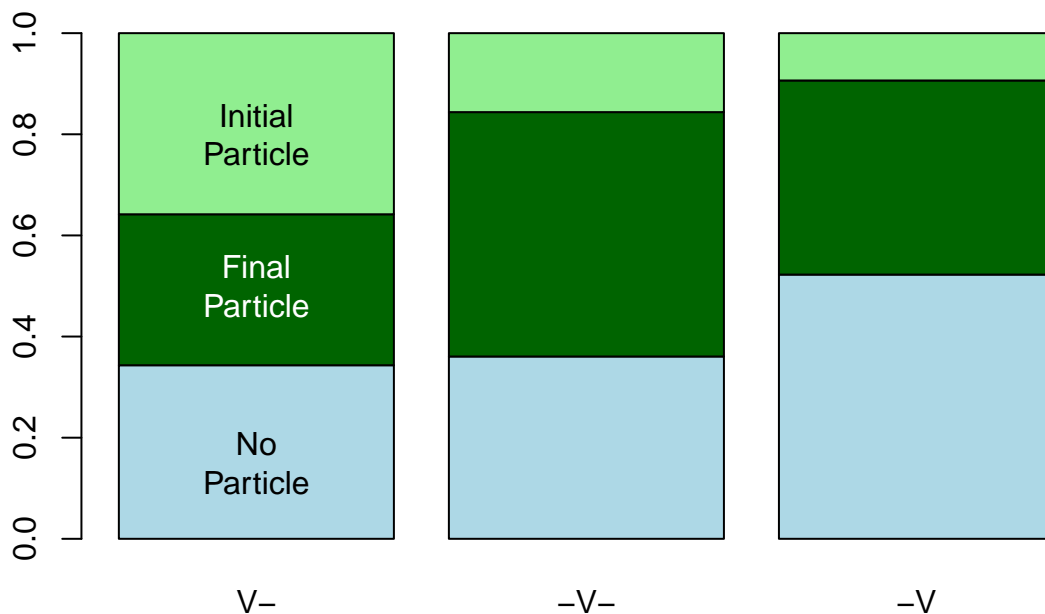
Show the raw data. This is just the number of languages with each type, uncorrected for historical relations.

```
table(d.wals$QP.pos3, d.wals$VerbPos)
```

```
##
##           1   2   3
##  1 None    23  97 162
##  2 Final    20 130 119
##  3 Initial   24  42  29
```

Plot the raw data.

```
#
tx4 = table(d.wals$QP.pos3,d.wals$VerbPos)
barplot(
  t(t(tx4)/colSums(tx4)),
  names.arg=c("V-", "-V-", "-V"),
  col=rev(c("light green", "dark green", "light blue")))
text(0.7, 0.8, "Initial\nParticle")
text(0.7, 0.5, "Final\nParticle", col='white')
text(0.7, 0.15, "No\nParticle")
```



Mixed effects models

Make three mixed effects models, each predicting the probability of a verb being in a given position by the position of the polar question marker. There is a random effect for families, which adds a basic control for historical relatedness.

```
lmVSO = glmer(VerbPos==1~QP.pos3+(1|family), data=d.wals, family=binomial(link=logit))
lmSVO = glmer(VerbPos==2~QP.pos3+(1|family), data=d.wals, family=binomial(link=logit))
lmSOV = glmer(VerbPos==3~QP.pos3+(1|family), data=d.wals, family=binomial(link=logit))
```

Use the fixed effects of the models to estimate the proportion of each question particle type for each basic word order type. This estimate should take into account the historical relations between families.

```
x = c(exp(fixef(lmVSO)[1]),exp(sum(fixef(lmVSO)[1:2])),exp(sum(fixef(lmVSO)[c(1,3)])))
vsoP = x/sum(x)
x = c(exp(fixef(lmSVO)[1]),exp(sum(fixef(lmSVO)[1:2])),exp(sum(fixef(lmSVO)[c(1,3)])))
svoP = x/sum(x)
x = c(exp(fixef(lmSOV)[1]),exp(sum(fixef(lmSOV)[1:2])),exp(sum(fixef(lmSOV)[c(1,3)])))
sovP = x/sum(x)
```

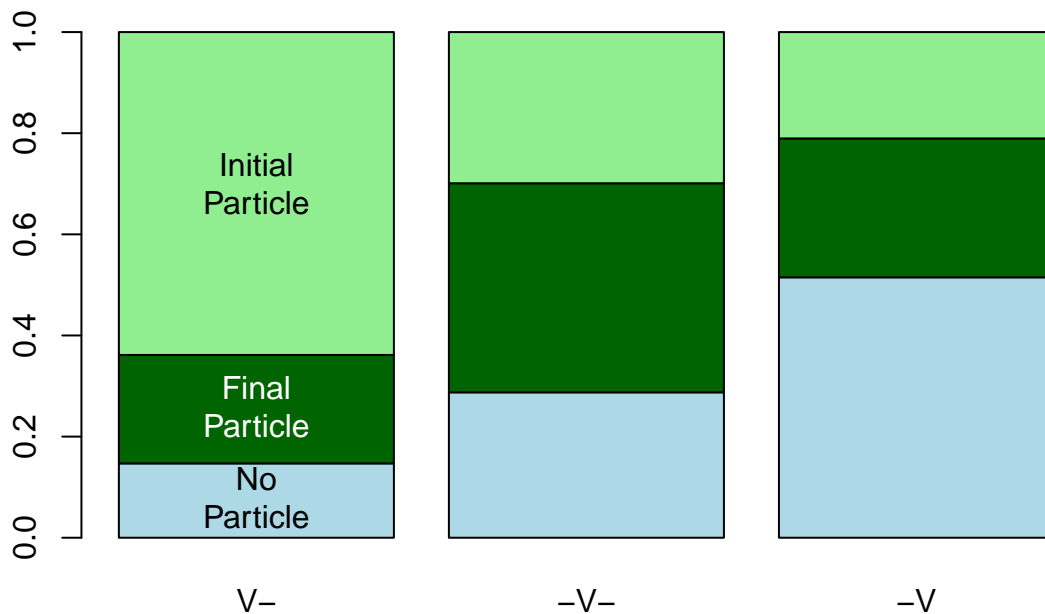
Show the model estimates.

```
estX = rbind(vsoP,svoP,sovP)
estX = estX/rowSums(estX)
colnames(estX) = c("None","Final","Initial")
estX
```

```
##           None      Final      Initial
## vsoP 0.1470058 0.2144506 0.6385436
## svoP 0.2875475 0.4135087 0.2989438
## sovP 0.5150098 0.2746864 0.2103037
```

Plot the model estimates. They reflect the raw estimates well.

```
barplot(
  t(estX),
  beside=F,
  names.arg=c("V-", "-V-", "-V"),
  col=rev(c("light green","dark green","light blue")))
text(0.7,0.7,"Initial\nParticle")
text(0.7,0.26,"Final\nParticle",col='white')
text(0.7,0.08,"No\nParticle")
```



Save the results

```
estX2 = t(estX)
rownames(estX2) = c('No Partcile', 'Final Particle', 'Initial Particle')
save(estX2, file='QParticleRealDataEstiamtes.rDat')
```