Cognitive influences in language evolution: Dutch data

Introduction

This is the model code for Monaghan & Roberts, "Cognitive influences in language evolution: Psycholinguistic predictors of loan word borrowing". It takes data from the WOLD database of borrowing for Dutch and tries to predict whether a word has been borrowed or not according to various psycholinguitic measures.

Load libraries

```
library(mgcv)
library(sjPlot)
library(lattice)
library(ggplot2)
library(gplots)
library(dplyr)
library(party)
library(lmtest)
library(gridExtra)
library(itsadug)
library(car)
library(caret)
library(scales)
library(binom)


logit2per = function(X){
  return(exp(X)/(1+exp(X)))
}

rescaleGam = function(px, n, xvar, xlab=""){
  y = logit2per(px[[n]]$fit)
  x = px[[n]]$x *attr(xvar,"scaled:scale") + attr(xvar,"scaled:center")
  se.upper = logit2per(px[[n]]$fit+px[[n]]$se)
  se.lower = logit2per(px[[n]]$fit-px[[n]]$se)
  dx = data.frame(x=x,y=y,ci.upper=se.upper,ci.lower=se.lower)
  plen = ggplot(dx, aes(x=x,y=y))+
    geom_ribbon(aes(ymin=ci.lower,ymax=ci.upper), alpha=0.3)+
    geom_line(size=1) +
    xlab(xlab)+
    ylab("Probability of borrowing")+
    coord_cartesian(ylim = c(0,1))
  return(plen)
}
```

Load data

The Dutch data is processed very similarly to the English data. The full process can be found in the `processing` folder, but here we just load the final prepared data frame:

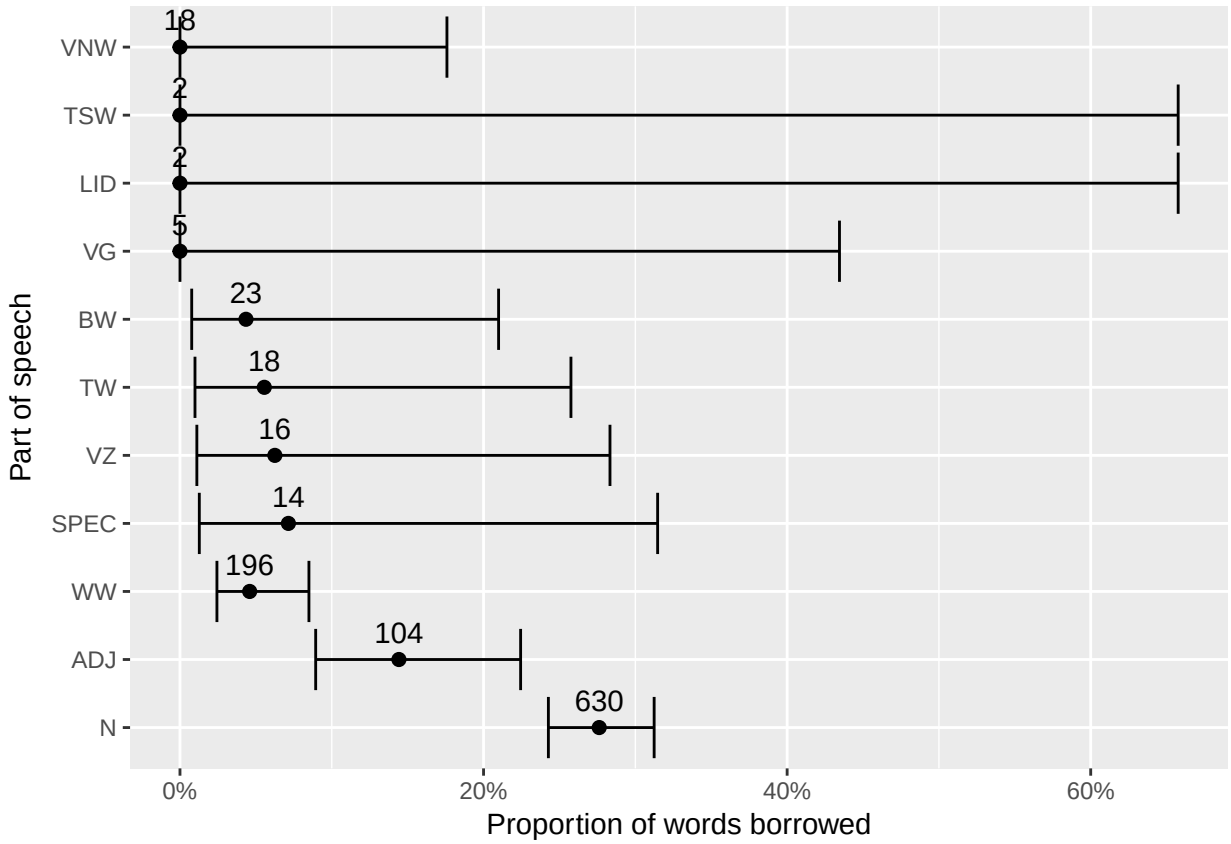
```
load("../data/loanwords_Dutch.Rdat")
```

Part of speech

We calculate the means, but the number of observations is very different for each category. We estimate confidence intervals around the mean with Wilson's binomial confidence interval method.

```
catx = data.frame(
  PoS = tapply(dutch$cat, dutch$cat, function(X){as.character(X[1])}),
  mean = tapply(dutch$bor15, dutch$cat, mean),
  n = tapply(dutch$bor15, dutch$cat, length),
  confint = binom.confint(
    tapply(dutch$bor15, dutch$cat, sum),
    tapply(dutch$bor15, dutch$cat, length),
    methods="wilson"
  )
)
catx = catx[order(catx$confint.lower, decreasing = T),]
catx$PoS = factor(catx$PoS, levels = catx[order(catx$confint.lower, decreasing = T),]$PoS)

posg = ggplot(catx, aes(x=mean, y=PoS)) +
  geom_point(size=2) +
  ylab("Part of speech") +
  xlab("Proportion of words borrowed")+
  scale_x_continuous(labels=percent_format()) +
  geom_text(aes(label=n), nudge_y=0.4) +
  geom_errorbarh(aes(xmin=confint.lower, xmax=confint.upper))
posg
```




```
pdf("../results/graphs/POS_Borrowing_Dutch.pdf",
    width = 6,
    height = 4)
posg
dev.off()
```

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

```
catx$mean= catx$mean*100
catx$confint.lower= catx$confint.lower*100
catx$confint.upper= catx$confint.upper*100
write.csv(catx[,c("PoS","mean",
                  'n','confint.lower','confint.upper')],
    "../results/Dutch_POS_BorrowingProportions.csv",
    row.names = F)
```

GAM model

Dutch data has 1028 datapoints.

The range of the length variable limits the number of knots that the gam model can fit:

```
m0.dutch = bam(bor15.cat ~
      s(phonlengthscale, k=3) +
      s(AoAscale) +
      s(subtlexzipfscale) +
```

```
      s(concscale) +
      s(cat,bs='re')+
      s(cat,phonlengthscale,bs='re')+
      s(cat,AoAscale,bs='re')+
      s(cat,subtlexzipfscale,bs='re')+
      s(cat,concscale,bs='re'),
    data = dutch,
    family='binomial')
```

```
summary(m0.dutch)
```

```
##
## Family: binomial
## Link function: logit
##
## Formula:
## bor15.cat ~ s(phonlengthscale, k = 3) + s(AoAscale) + s(subtlexzipfscale) +
##     s(concscale) + s(cat, bs = "re") + s(cat, phonlengthscale,
##     bs = "re") + s(cat, AoAscale, bs = "re") + s(cat, subtlexzipfscale,
##     bs = "re") + s(cat, concscale, bs = "re")
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -2.3389     0.3779  -6.189 6.04e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                               edf Ref.df Chi.sq  p-value
## s(phonlengthscale)      1.942e+00  1.996 18.147 0.000149 ***
## s(AoAscale)             1.310e+00  1.561 11.260 0.005448 **
## s(subtlexzipfscale)     3.630e+00  4.559 11.863 0.022505 *
## s(concscale)            1.663e+00  2.060  2.980 0.241790
## s(cat)                  3.724e+00 10.000 39.054 1.98e-08 ***
## s(cat,phonlengthscale)  1.721e-01 10.000  0.194 0.293757
## s(cat,AoAscale)         9.855e-06 10.000  0.000 0.938779
## s(cat,subtlexzipfscale) 1.883e-05 10.000  0.000 0.669506
## s(cat,concscale)        1.184e+00 10.000  3.069 0.098963 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.112   Deviance explained = 13.4%
## fREML =   1435  Scale est. = 1         n = 1028
```

Interactions

Test whether an interaction between AoA and frequency is warranted:

```
m1.dutch = bam(bor15.cat ~
      s(phonlengthscale, k=3) +
      s(AoAscale) +
      s(subtlexzipfscale) +
      s(concscale) +
      s(cat,bs='re')+
      s(cat,phonlengthscale,bs='re')+
      s(cat,AoAscale,bs='re')+
      s(cat,subtlexzipfscale,bs='re')+
      s(cat,concscale,bs='re') +
      te(AoAscale,subtlexzipfscale),
    data = dutch,
    family='binomial')

lrtest(m0.dutch,m1.dutch)
```

```
## Likelihood ratio test
##
## Model 1: bor15.cat ~ s(phonlengthscale, k = 3) + s(AoAscale) + s(subtlexzipfscale) +
##     s(concscale) + s(cat, bs = "re") + s(cat, phonlengthscale,
##     bs = "re") + s(cat, AoAscale, bs = "re") + s(cat, subtlexzipfscale,
##     bs = "re") + s(cat, concscale, bs = "re")
## Model 2: bor15.cat ~ s(phonlengthscale, k = 3) + s(AoAscale) + s(subtlexzipfscale) +
##     s(concscale) + s(cat, bs = "re") + s(cat, phonlengthscale,
##     bs = "re") + s(cat, AoAscale, bs = "re") + s(cat, subtlexzipfscale,
##     bs = "re") + s(cat, concscale, bs = "re") + te(AoAscale,
##     subtlexzipfscale)
##      #Df  LogLik     Df  Chisq Pr(>Chisq)
## 1 18.472 -441.31
## 2 20.348 -439.66 1.8758 3.2968     0.1924
```

No significant improvement.

Test whether an interaction between AoA and length is warranted:

```
m2.dutch = bam(bor15.cat ~
      s(phonlengthscale, k=3) +
      s(AoAscale) +
      s(subtlexzipfscale) +
      s(concscale) +
      s(cat,bs='re')+
      s(cat,phonlengthscale,bs='re')+
      s(cat,AoAscale,bs='re')+
      s(cat,subtlexzipfscale,bs='re')+
      s(cat,concscale,bs='re') +
      te(AoAscale,phonlengthscale),
    data = dutch,
    family='binomial')

lrtest(m0.dutch,m2.dutch)
```

```
## Likelihood ratio test
```

```
##
## Model 1: bor15.cat ~ s(phonlengthscale, k = 3) + s(AoAscale) + s(subtlexzipfscale) +
##     s(concscale) + s(cat, bs = "re") + s(cat, phonlengthscale,
##     bs = "re") + s(cat, AoAscale, bs = "re") + s(cat, subtlexzipfscale,
##     bs = "re") + s(cat, concscale, bs = "re")
## Model 2: bor15.cat ~ s(phonlengthscale, k = 3) + s(AoAscale) + s(subtlexzipfscale) +
##     s(concscale) + s(cat, bs = "re") + s(cat, phonlengthscale,
##     bs = "re") + s(cat, AoAscale, bs = "re") + s(cat, subtlexzipfscale,
##     bs = "re") + s(cat, concscale, bs = "re") + te(AoAscale,
##     phonlengthscale)
##      #Df  LogLik     Df  Chisq Pr(>Chisq)
## 1 18.472 -441.31
## 2 21.220 -439.07 2.7477 4.4682     0.2151
```

There is no improvement in log likelihood.

Test whether an interaction between Frequency and length is warranted:

```
m3.dutch = bam(bor15.cat ~
      s(phonlengthscale, k=3) +
      s(AoAscale) +
      s(subtlexzipfscale) +
      s(concscale) +
      s(cat,bs='re')+
      s(cat,phonlengthscale,bs='re')+
      s(cat,AoAscale,bs='re')+
      s(cat,subtlexzipfscale,bs='re')+
      s(cat,concscale,bs='re') +
      te(subtlexzipfscale,phonlengthscale),
    data = dutch,
    family='binomial')

lrtest(m0.dutch,m3.dutch)
```

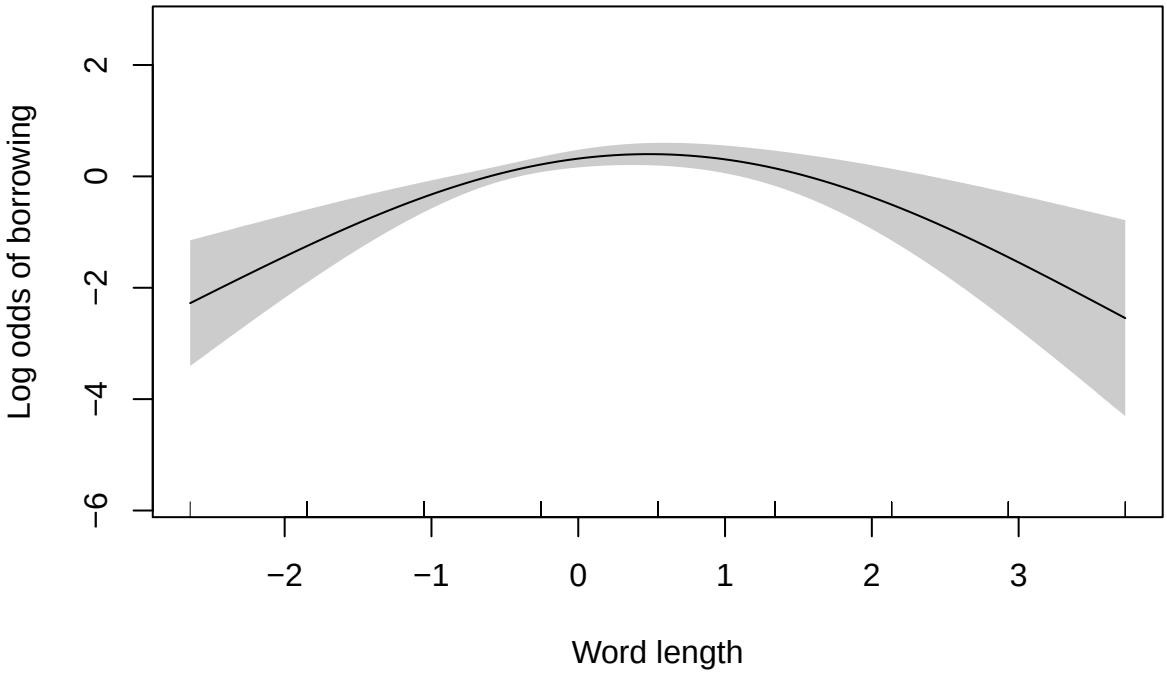
```
## Likelihood ratio test
##
## Model 1: bor15.cat ~ s(phonlengthscale, k = 3) + s(AoAscale) + s(subtlexzipfscale) +
##     s(concscale) + s(cat, bs = "re") + s(cat, phonlengthscale,
##     bs = "re") + s(cat, AoAscale, bs = "re") + s(cat, subtlexzipfscale,
##     bs = "re") + s(cat, concscale, bs = "re")
## Model 2: bor15.cat ~ s(phonlengthscale, k = 3) + s(AoAscale) + s(subtlexzipfscale) +
##     s(concscale) + s(cat, bs = "re") + s(cat, phonlengthscale,
##     bs = "re") + s(cat, AoAscale, bs = "re") + s(cat, subtlexzipfscale,
##     bs = "re") + s(cat, concscale, bs = "re") + te(subtlexzipfscale,
##     phonlengthscale)
##      #Df  LogLik     Df  Chisq Pr(>Chisq)
## 1 18.472 -441.31
## 2 22.040 -437.60 3.5681 7.4151     0.1155
```

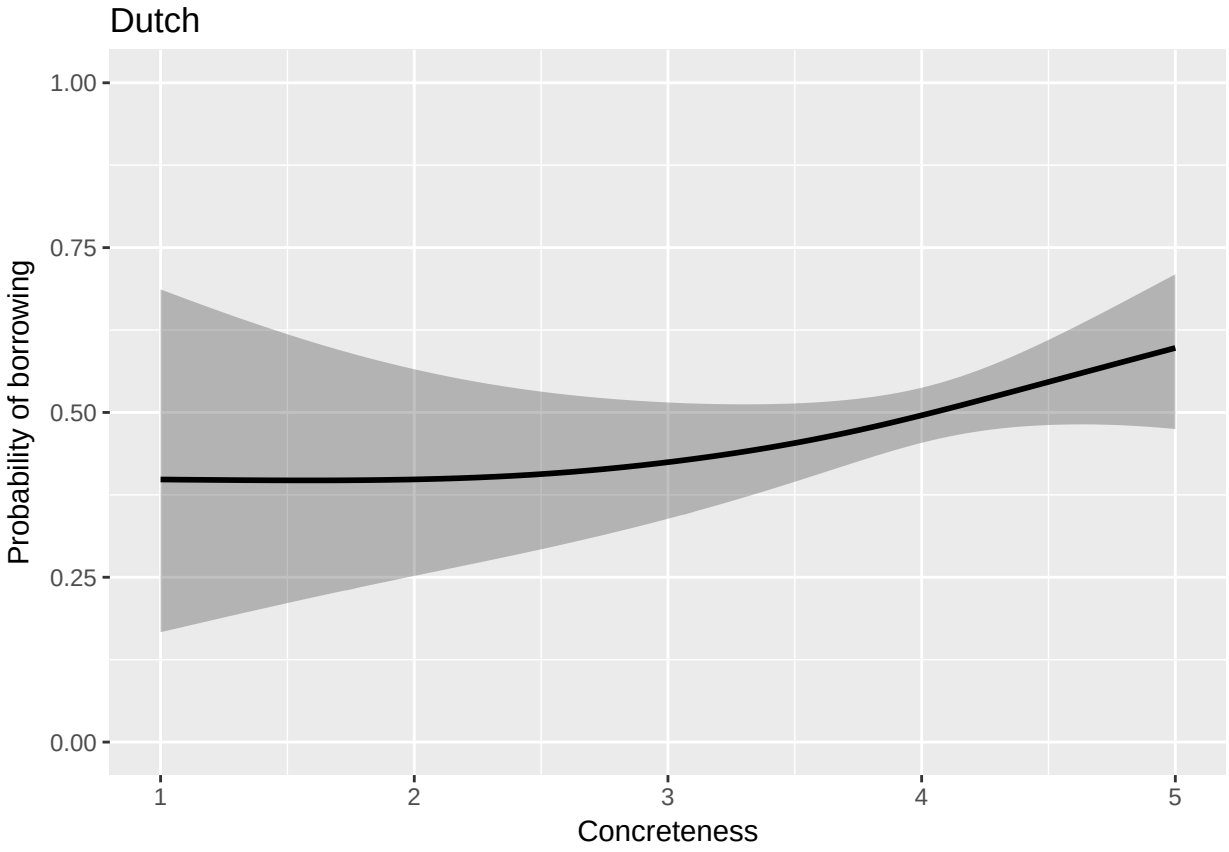
No significant improvement.

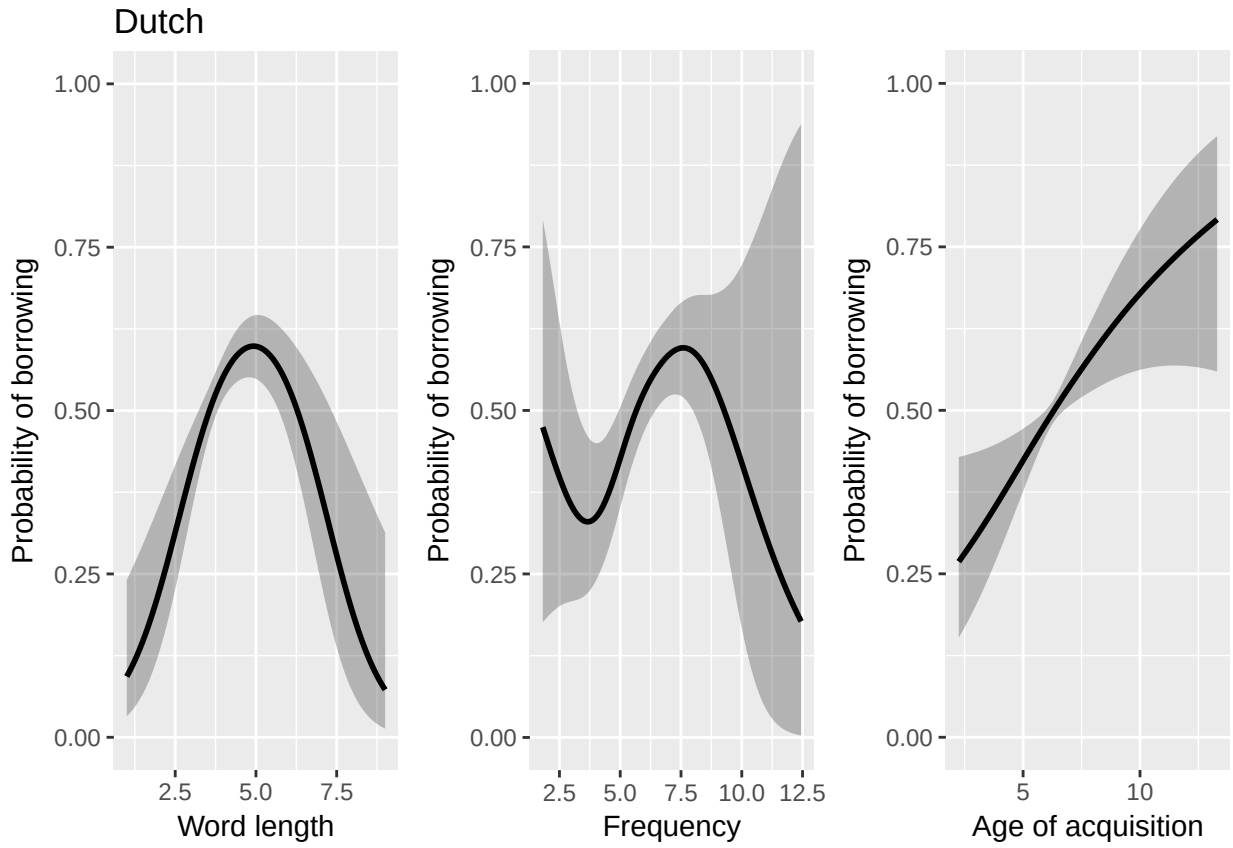
So no interactions are necessary.

Model estiamtes

Plot the model estimates, changing the dependent scale to probability and the independent variables to their original scales (code is hidden, but available in the Rmd file).

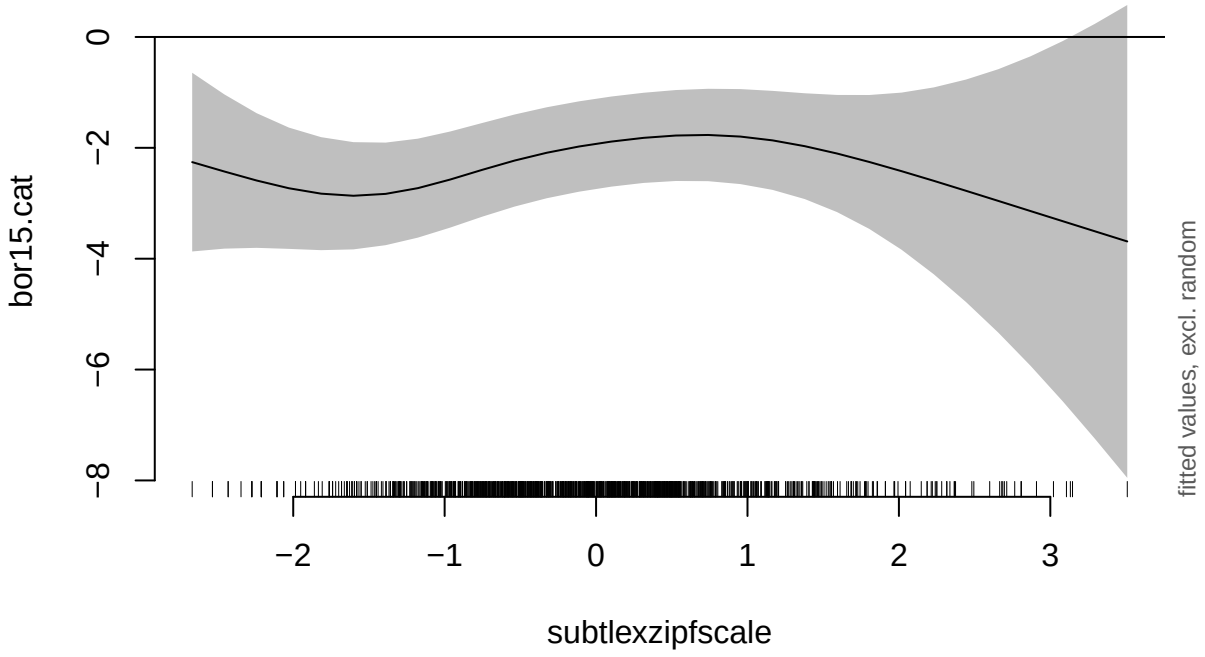


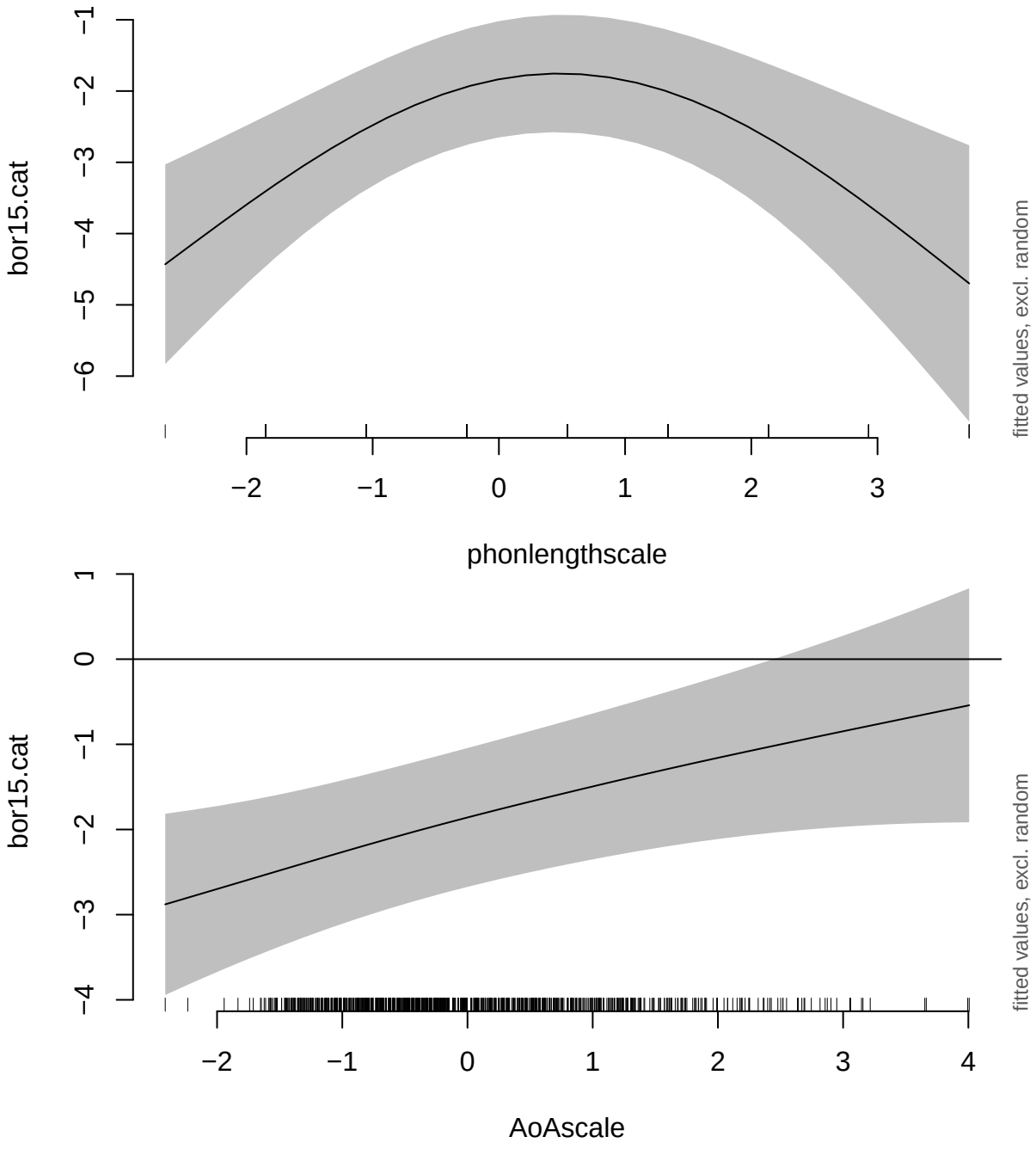


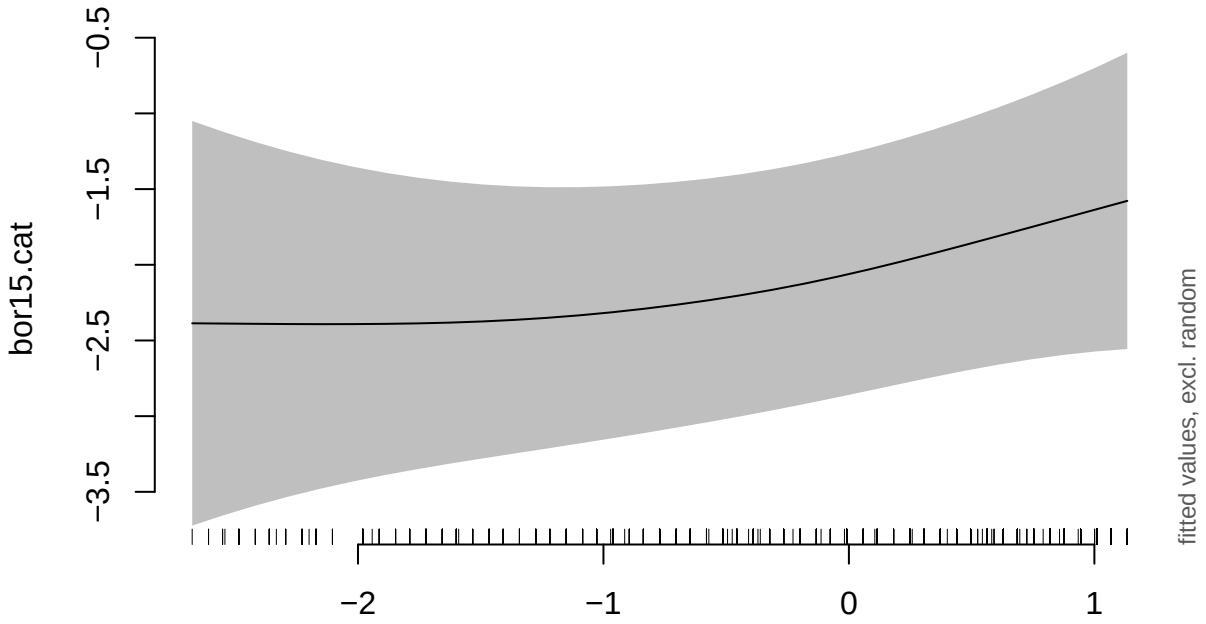


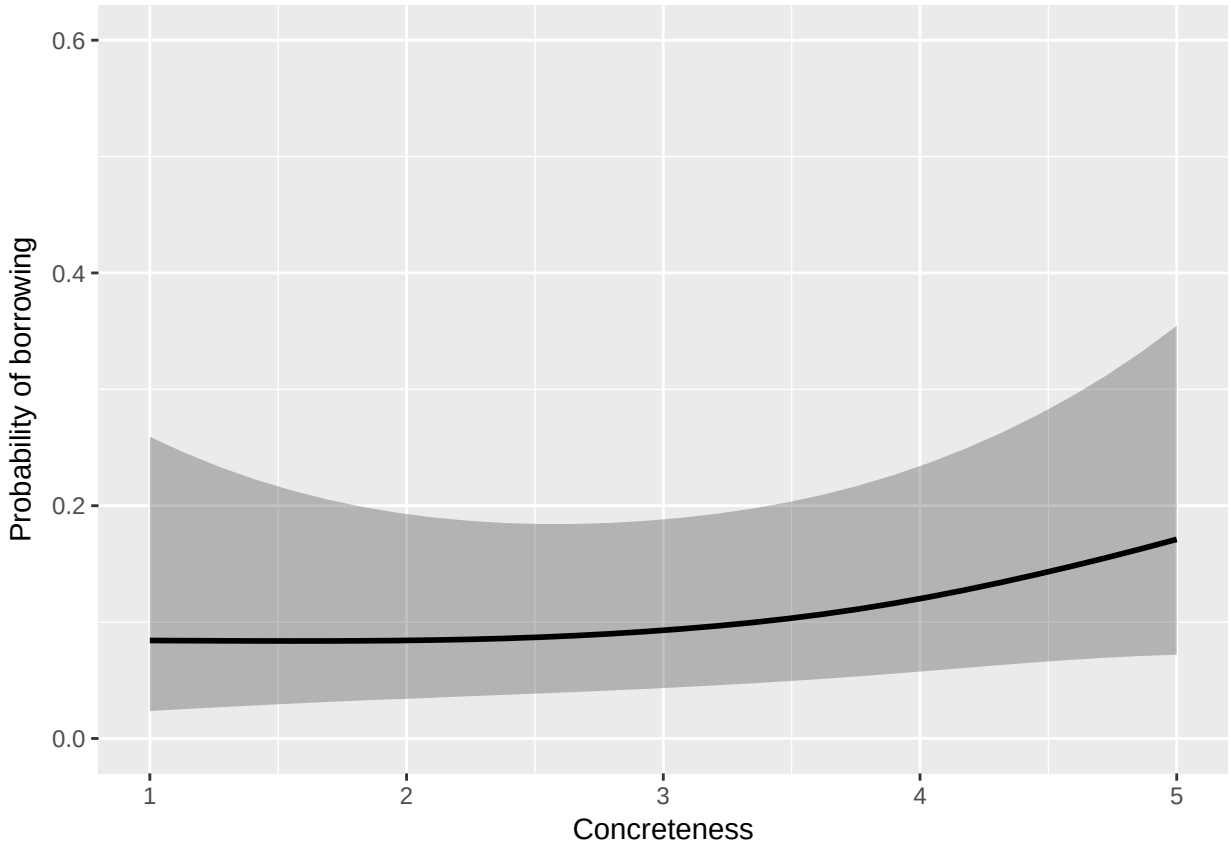

```
## pdf
##   2
```

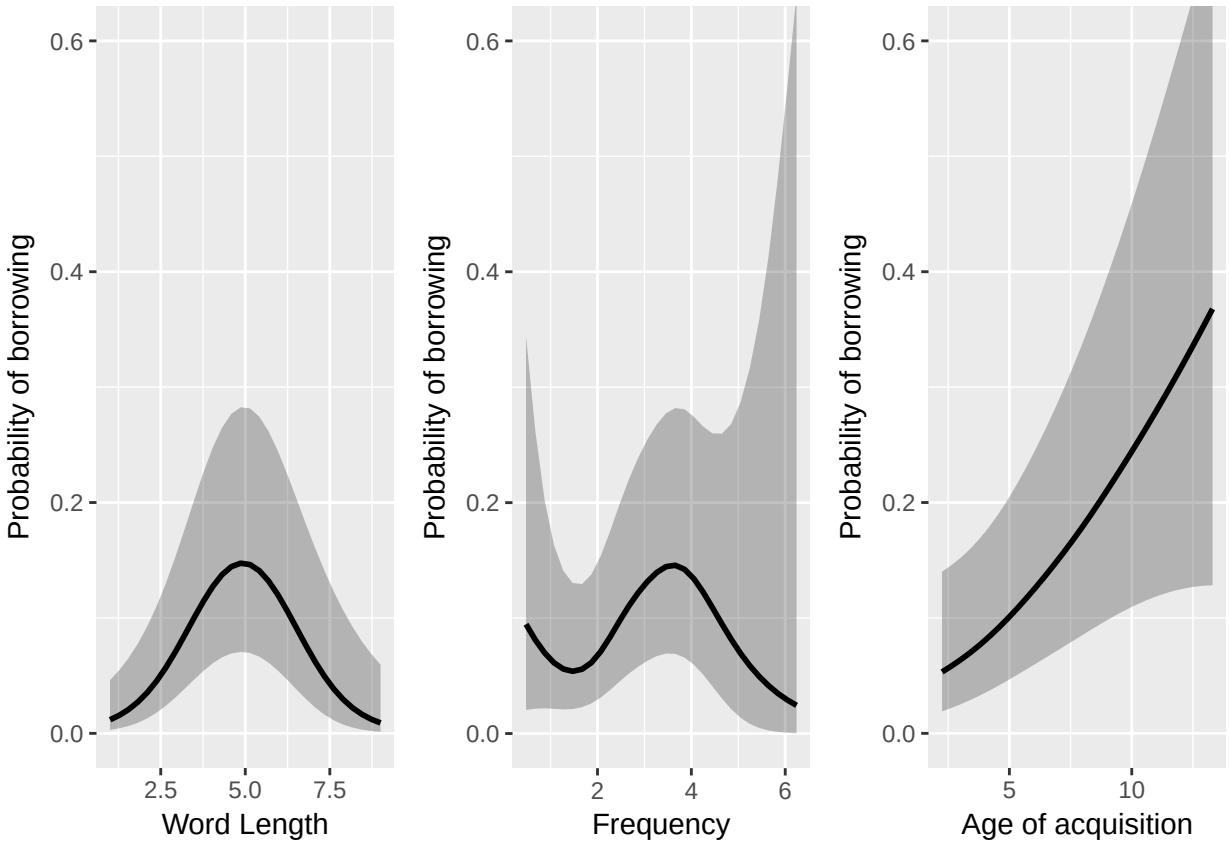
Plot the model estimates, removing the influence of the random effects using the library `itsadug` (code is hidden, but available in the Rmd file).




bor15.cat
−1
−2
−3
−4
−5
−6
−2
−1
0
1
2
3
phonlengthscale
fitted values, excl. random
bor15.cat
1
0
−1
−2
−3
−4
−2
−1
0
1
2
3
4
AoAscale
fitted values, excl. random


bor15.cat
−0.5
−1.5
−2.5
−3.5
−2
−1
0
1
concscale
fitted values, excl. random


Probability of borrowing
0.6
0.4
0.2
0.0
1
2
3
4
5
Concreteness




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

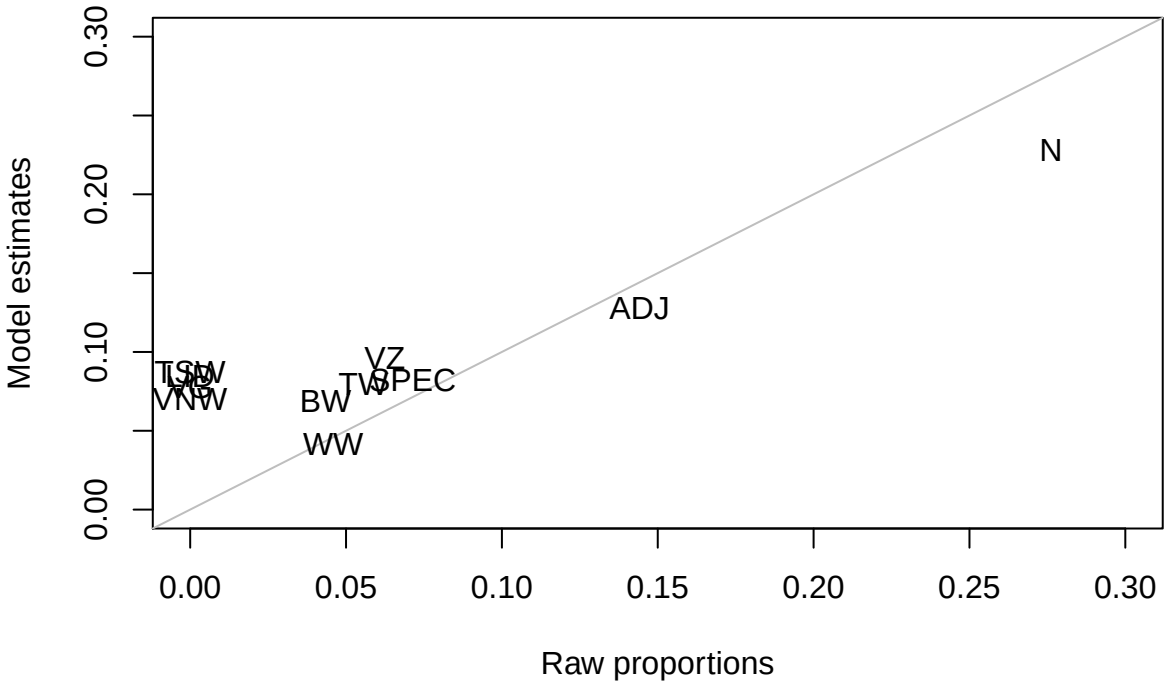
Random effects for Part of speech

```
mc = m0.dutch$coefficients
mc[grepl("s\\(cat\\)",names(mc))]
```

```
##     s(cat).1     s(cat).2     s(cat).3     s(cat).4     s(cat).5
##  1.119986827  0.421313755 -0.262839333 -0.037142605 -0.066124188
##     s(cat).6     s(cat).7     s(cat).8     s(cat).9    s(cat).10
## -0.004943456 -0.103771803 -0.131386158 -0.241569599  0.101695214
##    s(cat).11
## -0.795218653
```

```
raw = tapply(dutch$bor15,dutch$cat,mean)
model.est = logit2per(m0.dutch$coefficients[1] +
              mc[grepl("s\\(cat\\)",names(mc))])

plot(raw, model.est,
  xlab="Raw proportions",
  ylab="Model estimates",
  col="white",
     ylim=c(0,0.3),
     xlim=c(0,0.3))
abline(0,1,col='gray')
text(raw, model.est, names(raw))
```


Model estimates
0.00
0.10
0.20
0.30
N
ADJ
VZ
SPEC
TW
BW
WW
0.00
0.05
0.10
0.15
0.20
0.25
0.30
Raw proportions