Cognitive influences in language evolution: Study 4

Contents

Load libraries

```
library(mgcv)
library(sjPlot)
library(lattice)
library(ggplot2)
library(dplyr)
library(party)
library(lmtest)
library(gridExtra)
library(scales)
library(itsadug)
library(ggfortify)
library(factoextra)
library(gridExtra)
library(reshape2)
library(binom)
```

Extra helper functions:

```
source("GAM_derivaties.R")
logit2per = function(X){
  return(exp(X)/(1+exp(X)))
}

rescaleGam = function(px, n, xvar, xlab="",breaks=NULL,xlim=NULL){
  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")
  if(!is.null(breaks)){
    plen = plen + scale_x_continuous(breaks = breaks)
  }
  if(!is.null(xlim)){
   plen = plen + coord_cartesian(ylim = c(0,1),xlim=xlim)
  } else{
    plen = plen + coord_cartesian(ylim = c(0,1))
  }
  return(plen)
}
```

Load data

```
dataloan <- read.csv("../data/loanword10.csv",stringsAsFactors = F)
dataloan$bor15 <- ifelse(dataloan$borrowing==1,1, ifelse(dataloan$borrowing==5,0,NA))
dataloan$bor15.cat <- factor(dataloan$bor15)

dataloan$subtlexzipf = as.numeric(dataloan$subtlexzipf)
```

```
## Warning: NAs introduced by coercion
```

```
dataloan$AoA = as.numeric(dataloan$AoA)
```

```
## Warning: NAs introduced by coercion
```

```
dataloan$conc = as.numeric(dataloan$conc)
```

```
## Warning: NAs introduced by coercion
```

```
dataloan$old.english.length = as.numeric(dataloan$old.english.length)
aoaSD = sd(dataloan$AoA,na.rm = T)
aoaMean = mean(dataloan$AoA/aoaSD,na.rm=T)
dataloan$cat = factor(dataloan$cat)
dataloan$effect = factor(dataloan$effect)
```

Select only complete cases.

```
dataloan2 = dataloan[complete.cases(dataloan[,
               c("phonlength","AoA",
               "subtlexzipf", "cat",
               'conc','bor15')]),]
```

Scale and center:

```
dataloan2$AoAscale <- scale(dataloan2$AoA)
dataloan2$subtlexzipfscale <- scale(dataloan2$subtlexzipf)
phonlength.center = median(dataloan2$phonlength)
dataloan2$phonlengthscale <-
  dataloan2$phonlength - phonlength.center
phonlength.scale = sd(dataloan2$phonlengthscale)
dataloan2$phonlengthscale = dataloan2$phonlengthscale/phonlength.scale
```

```r
attr(dataloan2$phonlengthscale,"scaled:scale") = phonlength.scale
attr(dataloan2$phonlengthscale,"scaled:center") = phonlength.center
dataloan2$concscale <- scale(dataloan2$conc)
dataloan2$cat = relevel(dataloan2$cat,"Noun")
dataloan2$AoA_objscaled = scale(dataloan2$AoA_obj)
```

GAM for Old English

With old.english length as length. The idea of this analysis is to investigate whether length prior to borrowing affected whether the word was borrowed, rather than length of the borrowing. For that we need to go back to how the language was before the borrowings

Select only complete cases with old english length also:

```r
dataloan3 = dataloan2[complete.cases(dataloan2[,
                                              c("phonlength","AoA",
                                                "subtlexzipf", "cat",
                                                'conc','bor15','old.english.length')]),]
```

Scale length:

```r
old.english.length.center = median(dataloan3$old.english.length)
dataloan3$old.english.lengthscale <-
  dataloan3$old.english.length - old.english.length.center
old.english.length.scale = sd(dataloan3$old.english.lengthscale)
dataloan3$old.english.lengthscale = dataloan3$old.english.lengthscale/old.english.length.scale
attr(dataloan3$old.english.lengthscale,"scaled:scale") = old.english.length.scale
attr(dataloan3$old.english.lengthscale,"scaled:center") = old.english.length.center
```

Run GAM:

```r
m0oe = bam(bor15.cat ~
           s(old.english.lengthscale) +
           s(AoAscale) +
           s(subtlexzipfscale) +
           s(concscale) +
           s(cat,bs='re')+
           s(cat,old.english.lengthscale,bs='re')+
           s(cat,AoAscale,bs='re')+
           s(cat,subtlexzipfscale,bs='re')+
           s(cat,concscale,bs='re'),
         data = dataloan3,
         family='binomial')
summary(m0oe)
```

```
##
## Family: binomial
## Link function: logit
##
## Formula:
## bor15.cat ~ s(old.english.lengthscale) + s(AoAscale) + s(subtlexzipfscale) +
##     s(concscale) + s(cat, bs = "re") + s(cat, old.english.lengthscale,
##     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)   -1.673      0.449  -3.726 0.000195 ***
## ---
## 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(old.english.lengthscale)     1.000e+00  1.000  0.829 0.362568
## s(AoAscale)                    1.000e+00  1.000 37.763 7.99e-10 ***
## s(subtlexzipfscale)            3.641e+00  4.603 29.897 1.35e-05 ***
## s(concscale)                   3.001e+00  3.741 17.371 0.001776 **
## s(cat)                         5.253e+00 10.000 33.182 0.000708 ***
## s(cat,old.english.lengthscale) 2.366e+00 10.000 42.011 0.001721 **
## s(cat,AoAscale)                6.062e-06 10.000  0.000 0.726363
## s(cat,subtlexzipfscale)        1.558e-05 10.000  0.000 0.464271
## s(cat,concscale)               5.423e-06 10.000  0.000 0.831212
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.144   Deviance explained = 12.8%
## fREML = 1615.1  Scale est. = 1         n = 1139
```

There is a significant interaction between part of speech (`cat`) and old english length, but no significant effect of old english length on its own.

Test effect of old english length without interactions:

```
m1oe = bam(bor15.cat ~
             s(old.english.lengthscale) +
             s(AoAscale) +
             s(subtlexzipfscale) +
             s(concscale) +
             s(cat,bs='re'),
           data = dataloan3,
           family='binomial')
summary(m1oe)
```

```
##
## Family: binomial
## Link function: logit
##
## Formula:
## bor15.cat ~ s(old.english.lengthscale) + s(AoAscale) + s(subtlexzipfscale) +
##     s(concscale) + s(cat, bs = "re")
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -1.6669     0.4378  -3.808  0.00014 ***
## ---
## 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(old.english.lengthscale) 1.000  1.000  11.14 0.000843 ***
```

```
## s(AoAscale)                 1.000  1.000  38.49 5.52e-10 ***
## s(subtlexzipfscale)         3.649  4.615  29.56 1.62e-05 ***
## s(concscale)                2.710  3.384  15.14 0.003178 **
## s(cat)                      5.291 10.000  35.46 4.65e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.136   Deviance explained =   12%
## fREML = 1615.2  Scale est. = 1         n = 1139
```

Now Old English length is significant and linear, as for Modern English phonlength. But we are still including words that may have been borrowed BEFORE Old English, so select only those borrowed after Old English:

Words borrowed after Old English

Classify as borrowed only those words borrowed after 900CE (so have changed since old english)

```
dataloan3$age_oldest_num <- as.numeric(dataloan3$age_oldest_num)
dataloan3$bor15oe <- ifelse(dataloan3$bor15==1 & dataloan3$age_oldest_num <=1100 ,1, ifelse(dataloan3$bo
dataloan3$bor15oe.cat <- factor(dataloan3$bor15oe)
```

Run GAM:

```
m2oe = bam(bor15oe.cat ~
             s(old.english.lengthscale) +
             s(AoAscale) +
             s(subtlexzipfscale) +
             s(concscale) +
             s(cat,bs='re')+
             s(cat,old.english.lengthscale,bs='re')+
             s(cat,AoAscale,bs='re')+
             s(cat,subtlexzipfscale,bs='re')+
             s(cat,concscale,bs='re'),
           data = dataloan3,
           family='binomial')
summary(m2oe)
```

```
##
## Family: binomial
## Link function: logit
##
## Formula:
## bor15oe.cat ~ s(old.english.lengthscale) + s(AoAscale) + s(subtlexzipfscale) +
##     s(concscale) + s(cat, bs = "re") + s(cat, old.english.lengthscale,
##     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)  -1.7231     0.4485  -3.842 0.000122 ***
## ---
## 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(old.english.lengthscale)     1.000e+00  1.000  0.851 0.356390
## s(AoAscale)                    1.000e+00  1.000 33.304 7.88e-09 ***
## s(subtlexzipfscale)            3.497e+00  4.432 25.452 7.92e-05 ***
## s(concscale)                   2.569e+00  3.210 18.229 0.000617 ***
## s(cat)                         5.271e+00 10.000 32.640 0.000739 ***
## s(cat,old.english.lengthscale) 2.316e+00 10.000 41.004 0.001789 **
## s(cat,AoAscale)                2.586e-06 10.000  0.000 0.617430
## s(cat,subtlexzipfscale)        1.273e-05 10.000  0.000 0.456551
## s(cat,concscale)               3.120e-06 10.000  0.000 0.775858
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.141   Deviance explained = 12.7%
## fREML =   1583  Scale est. = 1         n = 1117
```

Again, interaction between cat and length, so remove interactions to investigate main effects:

```
m3oe = bam(bor15oe.cat ~
             s(old.english.lengthscale) +
             s(AoAscale) +
             s(subtlexzipfscale) +
             s(concscale) +
             s(cat,bs='re'),
           data = dataloan3,
           family='binomial')
summary(m3oe)
```

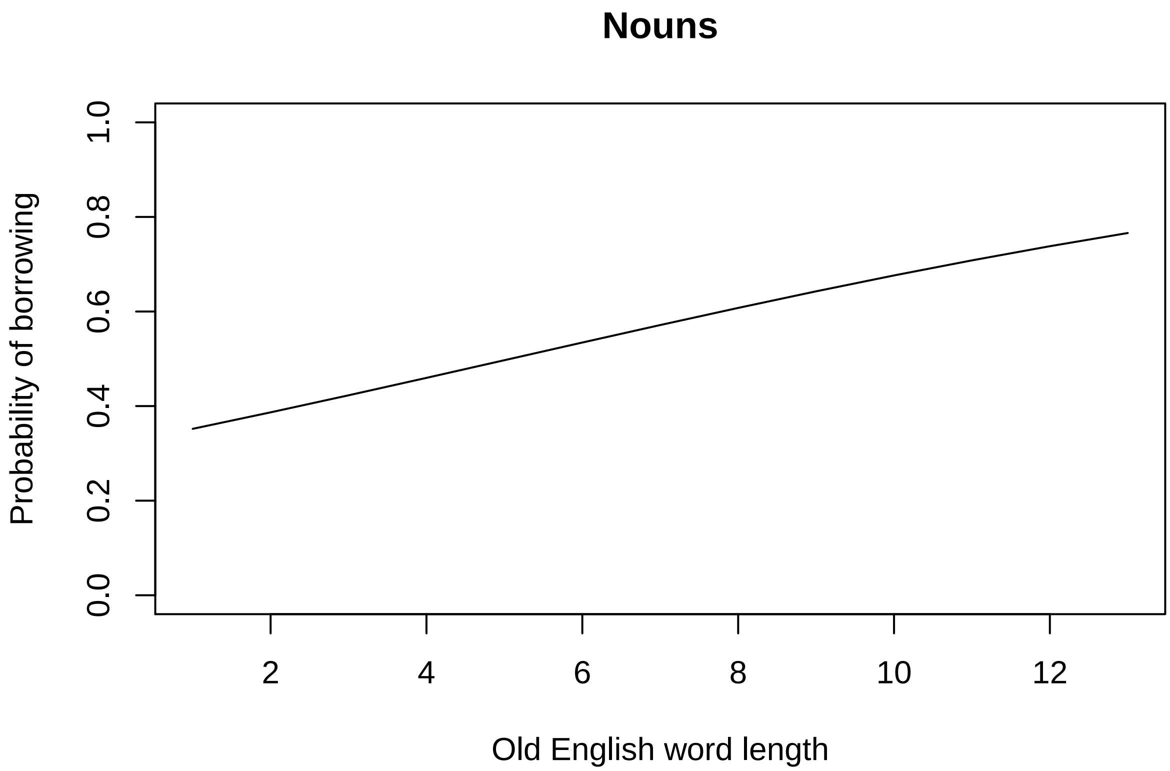
```
##
## Family: binomial
## Link function: logit
##
## Formula:
## bor15oe.cat ~ s(old.english.lengthscale) + s(AoAscale) + s(subtlexzipfscale) +
##     s(concscale) + s(cat, bs = "re")
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -1.7174     0.4372  -3.928 8.55e-05 ***
## ---
## 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(old.english.lengthscale) 1.000  1.000  11.27 0.000787 ***
## s(AoAscale)                1.000  1.000  33.90 5.82e-09 ***
## s(subtlexzipfscale)        3.502  4.440  25.09 9.56e-05 ***
## s(concscale)               2.227  2.789  14.28 0.001467 **
## s(cat)                     5.300 10.000  35.42 4.59e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.133   Deviance explained = 11.8%
## fREML = 1583.5  Scale est. = 1         n = 1117
```

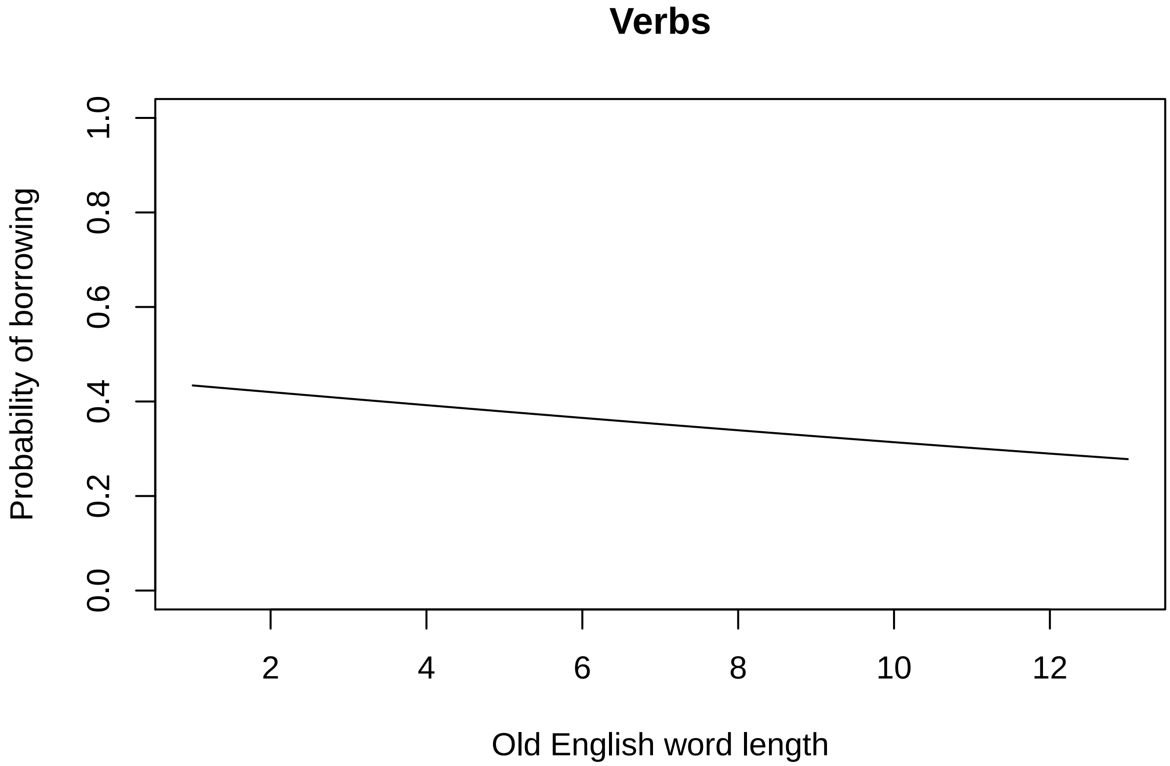
As before, a significant linear effect of length.

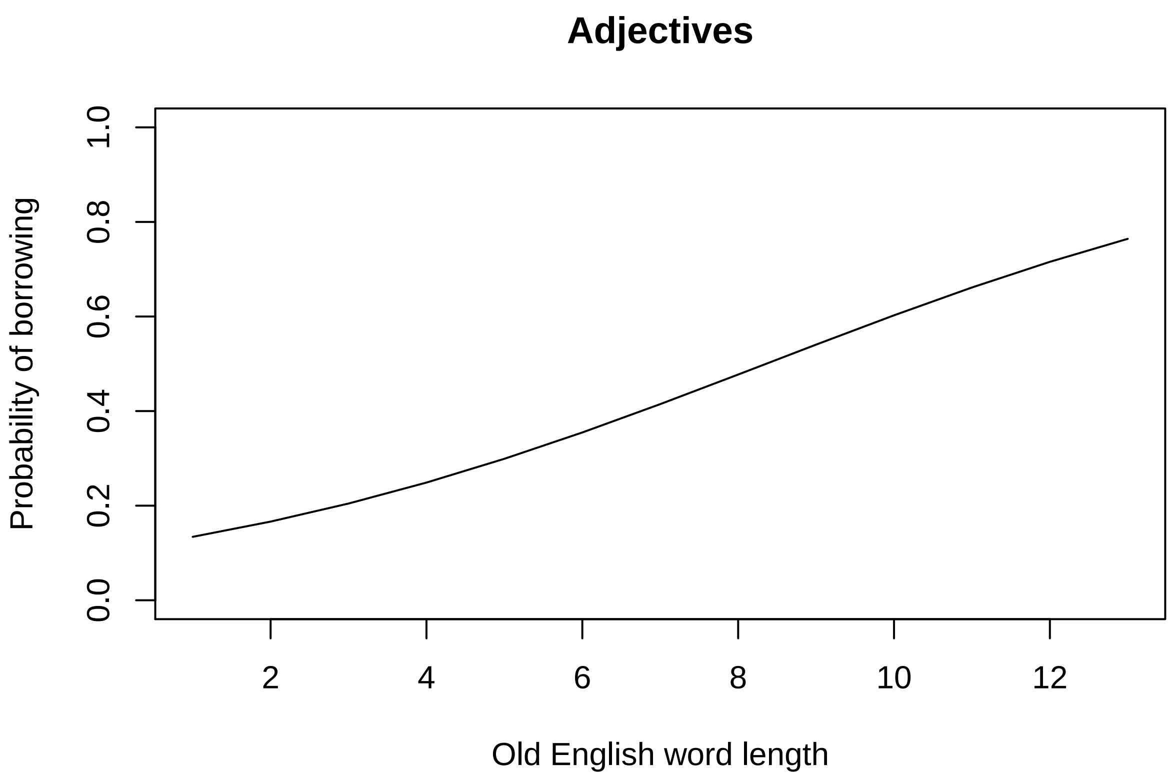
This is evidence that length of the Old English form influenced the likelihood of that word subsequently being borrowed, and is not just a consequence of borrowed words coming from languages with longer word length than English.

Effects for each part of speech

What is the relationship between old english word length and probability of borrowing for each part of speech? Below are plots which are calculated by using the model `m2oe` to predict borrowing probaility for a range of combinations of all variables, but keeping `cat` (part of speech) fixed to e.g. Nouns. Then the average borrowing probaility for each value of old english word length is plotted. The code is hidden, but can be seen in the Rmd file. These are for diagnosis only, formal tests are below.


Nouns
Probability of borrowing
0.0
0.2
0.4
0.6
0.8
1.0
2
4
6
8
10
12
Old English word length


Verbs
Probability of borrowing
0.0
0.2
0.4
0.6
0.8
1.0
2
4
6
8
10
12
Old English word length



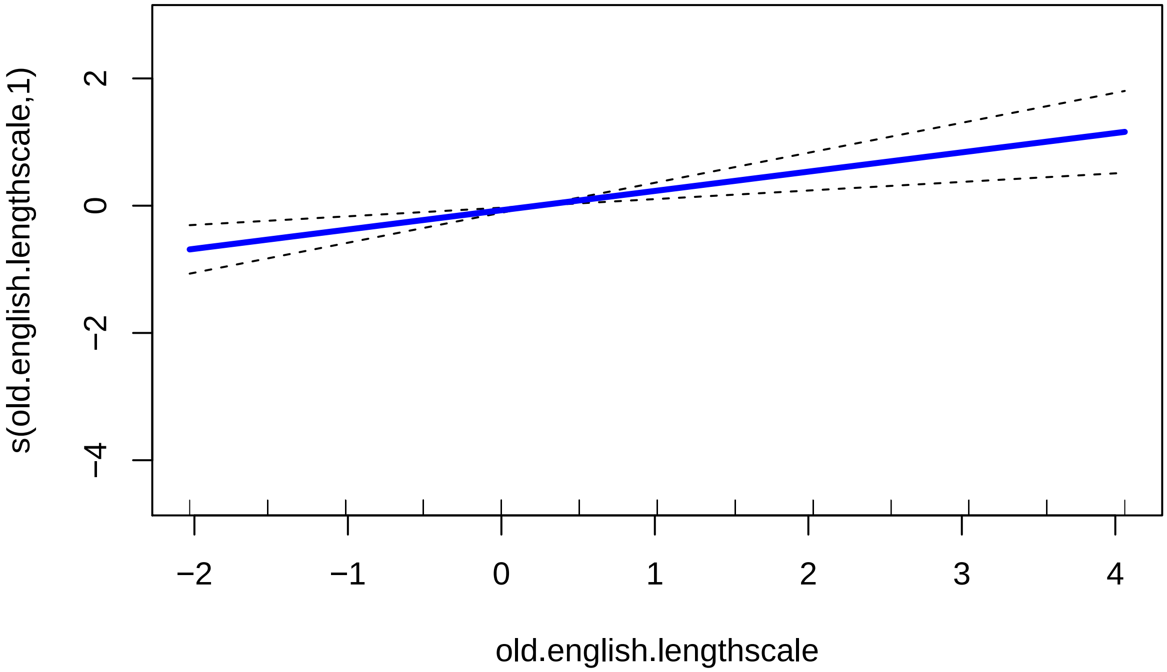
Analyse nouns only: Old English length is significant and monotonically increasing for nouns.

```
m2oen = bam(bor15oe.cat ~
              s(old.english.lengthscale) +
              s(AoAscale) +
              s(subtlexzipfscale) +
              s(concscale),
            data = dataloan3,subset = cat == "Noun",
            family='binomial')
summary(m2oen)
```

```
##
## Family: binomial
## Link function: logit
##
## Formula:
## bor15oe.cat ~ s(old.english.lengthscale) + s(AoAscale) + s(subtlexzipfscale) +
##     s(concscale)
##
## Parametric coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.35603    0.08483  -4.197 2.71e-05 ***
## ---
## 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(old.english.lengthscale) 1.000  1.000 13.015 0.000309 ***
## s(AoAscale)                1.535  1.910 21.068 7.13e-05 ***
## s(subtlexzipfscale)        2.943  3.770 16.431 0.002198 **
```

```
## s(concscale)                1.792  2.249  9.731 0.010753 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.108   Deviance explained = 9.02%
## fREML = 924.44  Scale est. = 1         n = 646
```

```
plotGAMSignificantSlopes(m2oen,"old.english.lengthscale","OE Length")
```



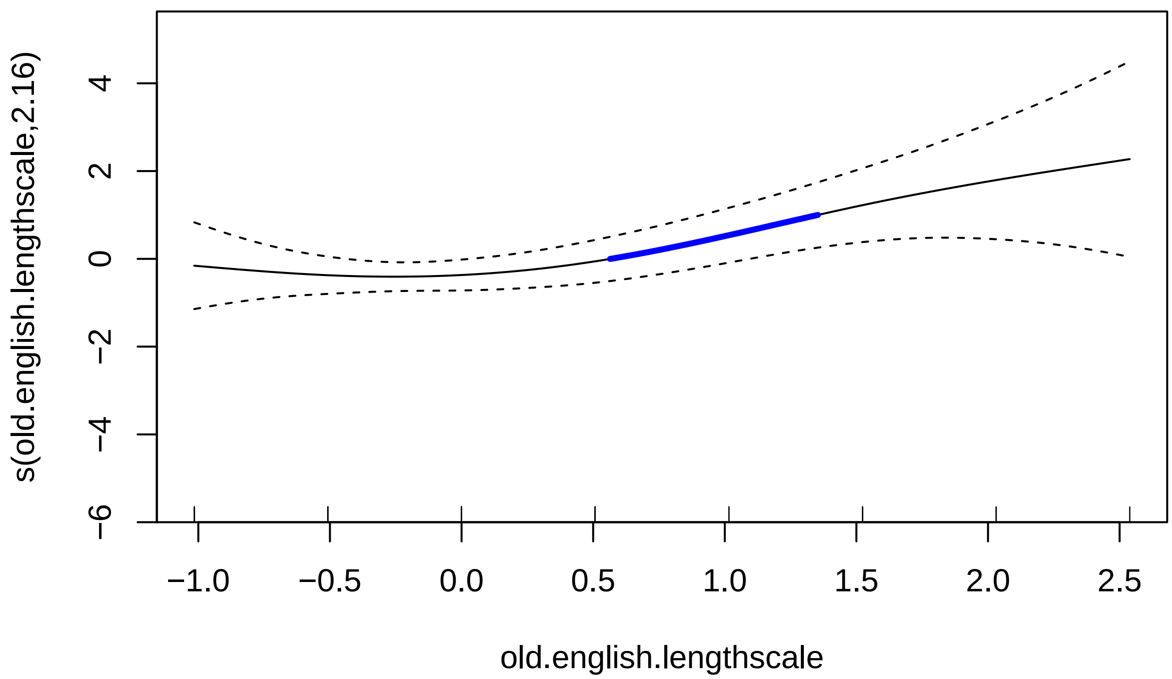
Analyse adjectives: Old English length is significant and increasing for adjectives.

```
m2oea = bam(bor15oe.cat ~
              s(old.english.lengthscale, k = 7) +
              s(AoAscale) +
              s(subtlexzipfscale) +
              s(concscale),
            data = dataloan3,subset = cat == "Adjective",
            family='binomial')
summary(m2oea)
```

```
##
## Family: binomial
## Link function: logit
##
## Formula:
## bor15oe.cat ~ s(old.english.lengthscale, k = 7) + s(AoAscale) +
##     s(subtlexzipfscale) + s(concscale)
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.8647     0.2249  -3.845 0.000121 ***
## ---
## 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(old.english.lengthscale) 2.164  2.725  8.311 0.02531 *
## s(AoAscale)                1.000  1.000  8.670 0.00323 **
## s(subtlexzipfscale)        1.293  1.529  4.059 0.05816 .
## s(concscale)               1.111  1.213  0.181 0.80144
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.168   Deviance explained = 16.9%
## fREML = 168.26  Scale est. = 1         n = 117
```

```
plotGAMSignificantSlopes(m2oea,"old.english.lengthscale","OE Length")
```



Analyse verbs: Length is not significantly related to the probability of borrowing for verbs.

```
m2oev = bam(bor15oe.cat ~
              s(old.english.lengthscale, k = 7) +
              s(AoAscale) +
              s(subtlexzipfscale) +
              s(concscale),
            data = dataloan3,subset = cat == "Verb",
            family='binomial')
summary(m2oev)
```

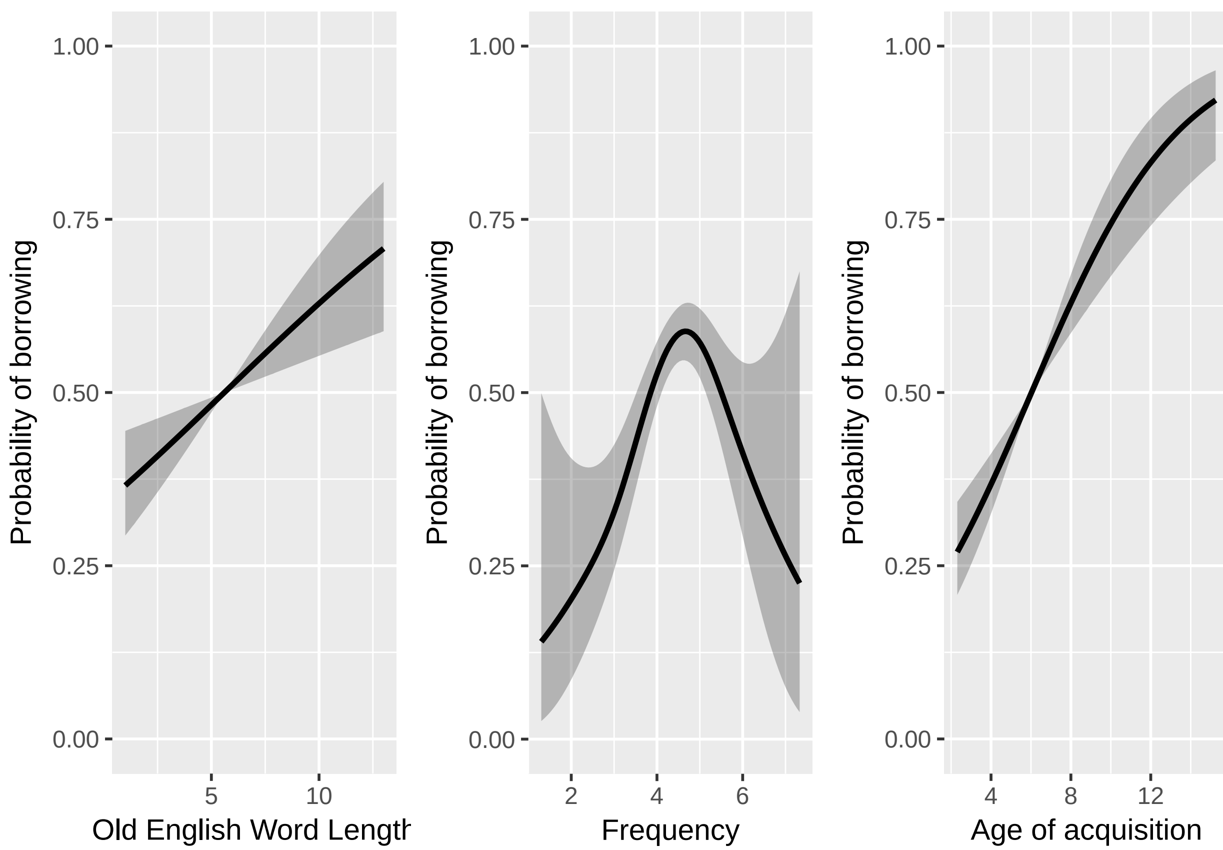
```
##
## Family: binomial
## Link function: logit
##
## Formula:
## bor15oe.cat ~ s(old.english.lengthscale, k = 7) + s(AoAscale) +
##     s(subtlexzipfscale) + s(concscale)
##
## Parametric coefficients:
```

```
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.7711     0.1387  -5.557 2.74e-08 ***
## ---
## 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(old.english.lengthscale) 2.480  3.132  4.088 0.23811
## s(AoAscale)                1.000  1.000  4.502 0.03385 *
## s(subtlexzipfscale)        3.102  3.938  6.174 0.20907
## s(concscale)               1.000  1.000  8.458 0.00363 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.125   Deviance explained = 11.6%
## fREML = 394.53  Scale est. = 1         n = 274
```

Model plots

Plot the model estimates with Old English length, changing the dependent scale to probability and the independent variables to their original scales. This code is hidden, but you can view it in the Rmd file.

Plotting model estimates:



Swadesh words

Next analyses are for likelihood of borrowing just for the Swadesh words:

Identify Swadesh words:

```
swd = read.csv("../data/SwadeshConcepts.txt",
               header = F, stringsAsFactors = F)$V1
dataloan3$Swadesh = dataloan3$word %in% swd
```

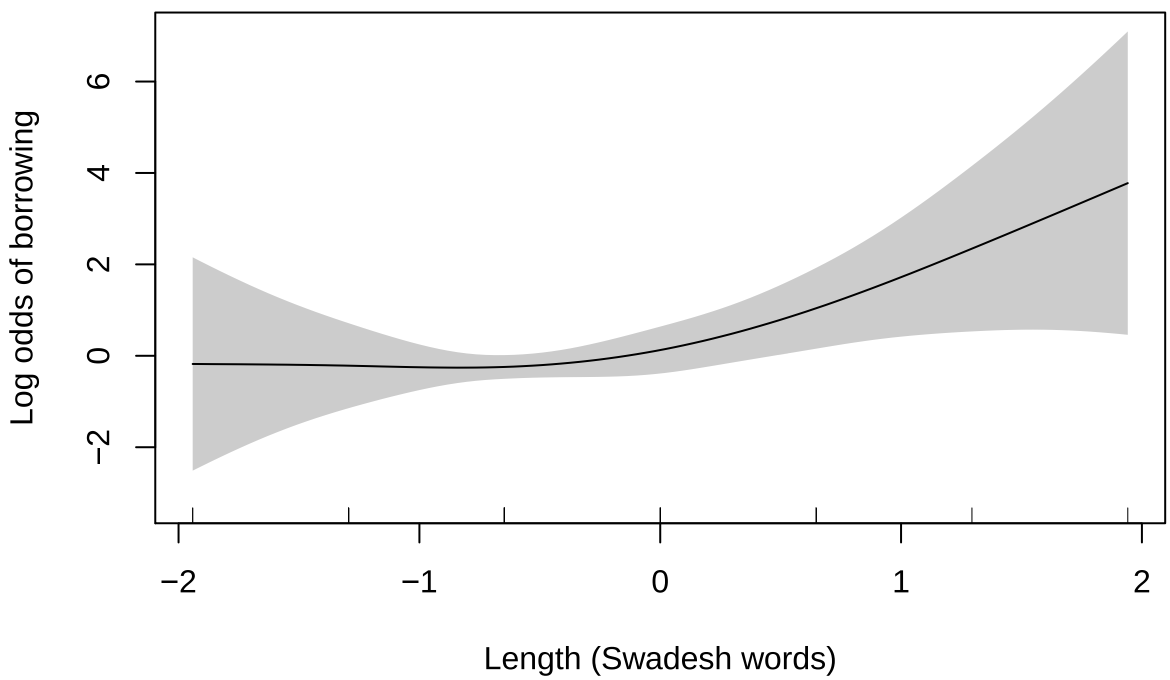
Run GAM:

```
m0sw = bam(bor15.cat ~
             s(phonlengthscale,k=7) +
             s(AoAscale) +
             s(subtlexzipfscale) +
             s(concscale) +
             s(cat,bs='re'),
           data = dataloan3,
           subset = Swadesh == TRUE,
           family='binomial')
```

> Previous comment: "this runs, with marginal (linear) effect of frequency - and shows high frequency less likely to be borrowed consistent with Pagel"

The only significant effect is length.

```
px = plot.gam(m0sw,select=1, xlab="Length (Swadesh words)", ylab="Log odds of borrowing",shade = T)
```



Coexistence, insertion or replacement

This works out influences on type of borrowing: coexistence, insertion, or replacement:

```
dataloan2$effect2 = dataloan2$effect
dataloan2$effect2[!dataloan2$effect2 %in%
      c("Insertion","Replacement","Coexistence")] = NA
dataloan2$effect2 = factor(dataloan2$effect2)
dataloan2$effect2 = as.numeric(dataloan2$effect2)-1
```

Multinomial GAM (predicting probability of each category given length, AoA, frequency, concreteness and PoS).

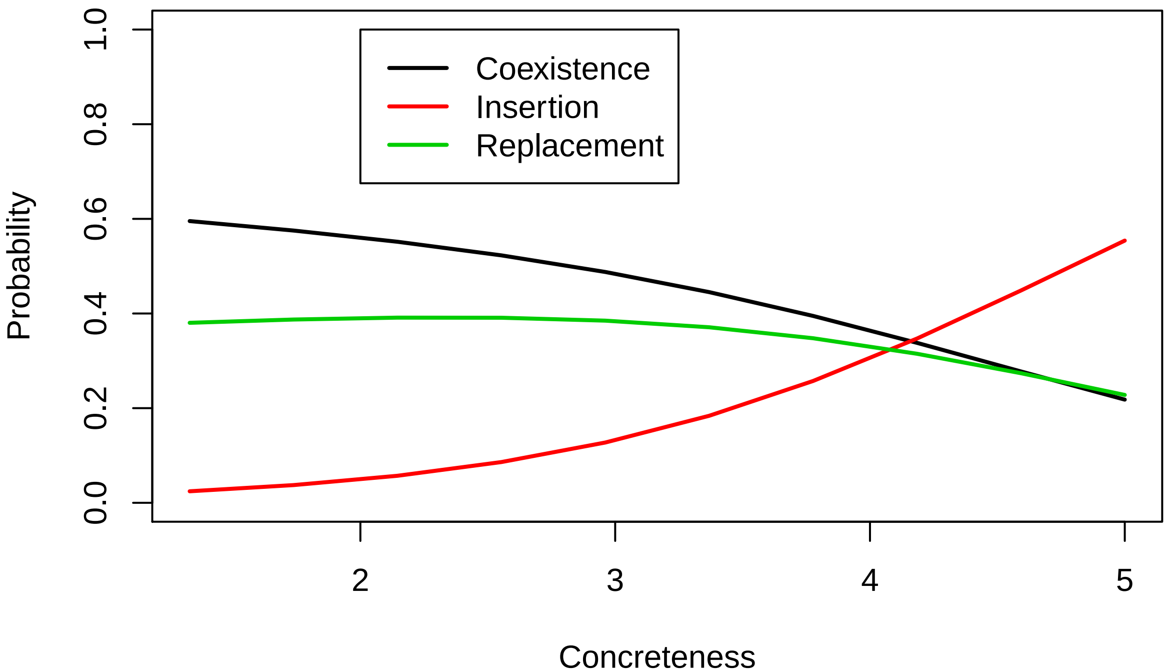
```
mCIR = gam(
        list(effect2 ~
            # formula for 2nd category
             s(phonlengthscale)+
             s(AoAscale) +
             s(subtlexzipfscale) +
             s(concscale) +
             s(cat,bs='re'),
            # formula for 3rd category
             ~ s(phonlengthscale)+
             s(AoAscale) +
             s(subtlexzipfscale) +
             s(concscale) +
             s(cat,bs='re')),
           data = dataloan2[!is.na(dataloan2$effect2),],
           family=multinom(K=2))
summary(mCIR)
```

```
##
## Family: multinom
## Link function:
##
## Formula:
## effect2 ~ s(phonlengthscale) + s(AoAscale) + s(subtlexzipfscale) +
##     s(concscale) + s(cat, bs = "re")
## ~s(phonlengthscale) + s(AoAscale) + s(subtlexzipfscale) + s(concscale) +
##     s(cat, bs = "re")
##
## Parametric coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -0.166581   0.298300  -0.558    0.577
## (Intercept).1 -0.007741   0.164043  -0.047    0.962
##
## Approximate significance of smooth terms:
##                         edf Ref.df Chi.sq  p-value
## s(phonlengthscale)    1.0001  1.000  0.175   0.6761
## s(AoAscale)           1.0003  1.001  0.573   0.4490
## s(subtlexzipfscale)   1.9658  2.526  1.946   0.4665
## s(concscale)          1.0138  1.027 39.430 4.18e-10 ***
## s(cat)                1.1957  6.000  3.325   0.0714 .
## s.1(phonlengthscale)  1.5543  1.939  2.154   0.3832
## s.1(AoAscale)         1.0001  1.000  0.030   0.8614
```

```
## s.1(subtlexzipfscale) 1.0001  1.000  0.047   0.8284
## s.1(concscale)        1.0000  1.000  0.934   0.3338
## s.1(cat)              0.5735  6.000  0.957   0.2024
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Deviance explained =   11%
## -REML =  332.4  Scale est. = 1         n = 477
```

Only concreteness is a significant predictor.

Visualise the relationship between concreteness and the probability of each category. Code is not shown, but available in the Rmd file.



Original separate binomial GAMs

```
dataloan2$insertornot =
  as.factor(dataloan2$effect=="Insertion")
dataloan2$coexistornot =
  as.factor(dataloan2$effect=="Coexistence")
dataloan2$replaceornot =
  as.factor(dataloan2$effect=="Replacement")
```

Run GAM:

```
m0insert = bam(insertornot ~
             s(phonlengthscale) +
             s(AoAscale) +
             s(subtlexzipfscale) +
             s(concscale) +
             s(cat,bs='re'),
           data = dataloan2,
           family='binomial')
summary(m0insert)
```

```
##
## Family: binomial
## Link function: logit
##
## Formula:
## insertornot ~ s(phonlengthscale) + s(AoAscale) + s(subtlexzipfscale) +
##     s(concscale) + s(cat, bs = "re")
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -2.647      0.308  -8.594   <2e-16 ***
## ---
## 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)  2.216  2.810 34.540 1.75e-07 ***
## s(AoAscale)         1.070  1.137 11.150  0.00149 **
## s(subtlexzipfscale) 2.501  3.215  8.500  0.04377 *
## s(concscale)        1.755  2.193 33.444 1.10e-07 ***
## s(cat)              1.850 11.000  7.381  0.00760 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.124   Deviance explained = 16.1%
## fREML = 1861.6  Scale est. = 1         n = 1302
```

> Previous comment: "This shows that insertion borrowings are more likely to be concrete than other types of borrowing. No other significant effects."

There are significant effects for length, AoA, frequency and concreteness.

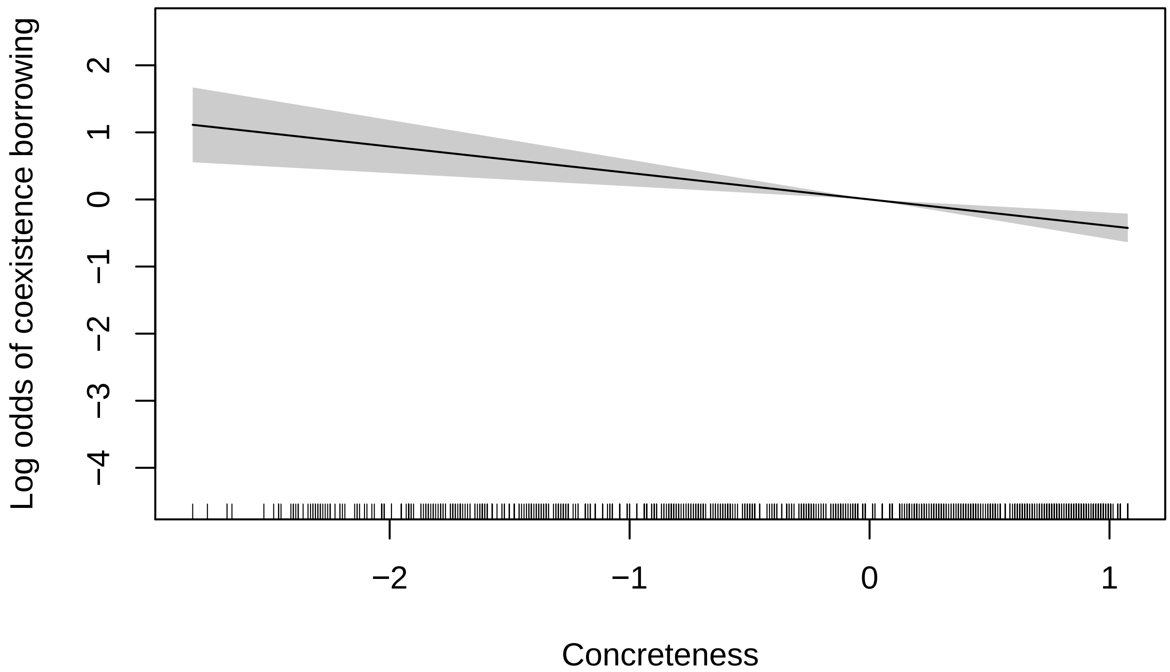
```
m0coexist = bam(coexistornot ~
                 s(phonlengthscale) +
                 s(AoAscale) +
```

```r
                 s(subtlexzipfscale) +
                 s(concscale) +
                 s(cat,bs='re'),
               data = dataloan2,
               family='binomial')
summary(m0coexist)
```

```
##
## Family: binomial
## Link function: logit
##
## Formula:
## coexistornot ~ s(phonlengthscale) + s(AoAscale) + s(subtlexzipfscale) +
##     s(concscale) + s(cat, bs = "re")
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -2.2557     0.1016   -22.2   <2e-16 ***
## ---
## 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)  1.000e+00  1.000 22.156 2.52e-06 ***
## s(AoAscale)         1.000e+00  1.000  5.298   0.0214 *
## s(subtlexzipfscale) 4.128e+00  5.162 12.912   0.0304 *
## s(concscale)        1.000e+00  1.000 15.900 6.68e-05 ***
## s(cat)              3.323e-06 11.000  0.000   0.4969
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.0692   Deviance explained =  8.7%
## fREML = 1843.3  Scale est. = 1         n = 1302
```

This shows that coexistence borrowings are more likely to be abstract than other types of borrowing, and affected by category.

```r
px = plot.gam(m0coexist,select=4, xlab="Concreteness", ylab="Log odds of coexistence borrowing",shade =
```




```r
m0replace = bam(replaceornot ~
                  s(phonlengthscale) +
                  s(AoAscale) +
                  s(subtlexzipfscale) +
                  s(concscale) +
                  s(cat,bs='re'),
                data = dataloan2,
                family='binomial')
summary(m0replace)
```

```
##
## Family: binomial
## Link function: logit
##
## Formula:
## replaceornot ~ s(phonlengthscale) + s(AoAscale) + s(subtlexzipfscale) +
##     s(concscale) + s(cat, bs = "re")
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -3.1638     0.4699  -6.733 1.66e-11 ***
## ---
## 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)  1.000  1.000  7.470 0.006273 **
## s(AoAscale)         1.000  1.000  2.055 0.151706
## s(subtlexzipfscale) 1.983  2.544  4.516 0.182908
## s(concscale)        1.000  1.000 11.068 0.000879 ***
## s(cat)              4.458 11.000 17.583 0.000440 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## R-sq.(adj) =  0.0363   Deviance explained = 5.77%
## fREML = 1825.8  Scale est. = 1         n = 1302
```

> Previous comment: "This shows that replacement borrowings are more likely to be abstract than other types of borrowing, and affected by category."

This shows that replacement borrowings are more likely to be abstract than other types of borrowing.

```
px = plot.gam(m0replace,select=4, xlab="Concreteness", ylab="Log odds of replacement borrowing",shade =
```