

## Data, data documentation and analysis scripts for

### *Quantitative Social Dialectology: Explaining Linguistic Variation Geographically and Socially*

Martijn Wieling<sup>(1,2)</sup> & John Nerbonne<sup>(1)</sup> & R. Harald Baayen<sup>(2,3)</sup>

<sup>1</sup>University of Groningen, the Netherlands & <sup>2</sup>Eberhard Karls University, Germany & <sup>3</sup>University of Alberta, Canada

Journal: **PLOS ONE**

Open access: <http://www.plosone.org/article/info:doi/10.1371/journal.pone.0023613>

## Abstract

In this study we examine linguistic variation and its dependence on both social and geographic factors. We follow dialectometry in applying a quantitative methodology and focusing on dialect distances, and social dialectology in the choice of factors we examine in building a model to predict word pronunciation distances from the standard Dutch language to 424 Dutch dialects. We combine linear mixed-effects regression modeling with generalized additive modeling to predict the pronunciation distance of 559 words. Although geographical position is the dominant predictor, several other factors emerged as significant. The model predicts a greater distance from the standard for smaller communities, for communities with a higher average age, for nouns (as contrasted with verbs and adjectives), for more frequent words, and for words with relatively many vowels. The impact of the demographic variables, however, varied from word to word. For a majority of words, larger, richer and younger communities are moving towards the standard. For a smaller minority of words, larger, richer and younger communities emerge as driving a change away from the standard. Similarly, the strength of the effects of word frequency and word category varied geographically. The peripheral areas of the Netherlands showed a greater distance from the standard for nouns (as opposed to verbs and adjectives) as well as for high-frequency words, compared to the more central areas. Our findings indicate that changes in pronunciation have been spreading (in particular for low-frequency words) from the Hollandic center of economic power to the peripheral areas of the country, meeting resistance that is stronger wherever, for well-documented historical reasons, the political influence of Holland was reduced. Our results are also consistent with the theory of lexical diffusion, in that distances from the Hollandic norm vary systematically and predictably on a word by word basis.

**Keywords:** dialectology, dialectometry, generalized additive modeling, mixed-effects regression, Dutch dialects.

# 1 Packages and functions

```
library(mgcv)
library(lme4.0) # results based on old lme4

R.Version()$version.string

## [1] "R version 3.0.2 (2013-09-25)"

packageVersion("mgcv")

## [1] '1.7.29'

packageVersion("lme4.0")

## [1] '0.999999.4'

source('functions/functions.R') # custom plotting and helper functions
```

## 2 Data set

```
load("data/dialectNL.rda")
```

Legenda dialectNL (225866 observations of 37 variables):

Note that the columns with a suffix of `.c` or `.z` are not described here. These are simply a centered (mean equals zero) or standardized (mean equals zero and standard deviation equals 1) version of the corresponding variables shown below.

1. Word : the word for which pronunciations were included
2. Transcriber : the transcriber of the subject's pronunciations
3. Location : the dialect location
4. PronDistStdDutch : Pronunciation distance from standard Dutch
5. Longitude : longitude of the dialect location
6. Latitude : latitude of the dialect location
7. Geo : the non-linear influence of geography (see below)
8. WordFreq.log : word frequency (log-transformed)
9. WordCategory : word category (Noun, Adjective, Verb, adverB)
10. WordIsNounOrAdverb : Contrast indicating if the word is a noun or adverb (1) or not (0)
11. WordLength.log : number of sounds in the standard pronunciation of the word (log-transformed)
12. WordVCratio.log : vowel-to-consonant ratio in the standard pronunciation of the word
13. PopSize.log : number of inhabitants in the location (log-transformed)
14. PopSize.log\_residGeo : as above, but excluding the influence of geography
15. PopAvgIncome.log : average income in the location (log-transformed)
16. PopAvgIncome.log\_residGeo : as above, but excluding the influence of geography
17. PopAvgAge : average age in the location
18. PopAvgAge\_residPopAvgIncome.log\_Geo : as above, but excluding the influence of average income and geography
19. PopMaleFemaleRatio : male-female ratio in the location
20. SpeakerIsMale : value between 0 and 1 indicating the proportion of speakers who are male (incidentally more speakers were present)

21. SpeakerBirthYear : year of birth of the speaker (or average when multiple speakers were present)
22. SpeakerEmploymentLevel : employment level of the speaker (or average when multiple speakers were present)
23. SpeakerRecordingYear : year in which the speaker was recorded
24. FieldworkerIsMale : value between 0 and 1 indicating the proportion of transcribers who are male (incidentally more transcribers were present)

### 3 Analysis and results

#### Representing geography with a generalized additive model

```
geo = gam(PronDistStdDutch.c ~ s(Longitude, Latitude), data=dialectNL)
summary(geo)

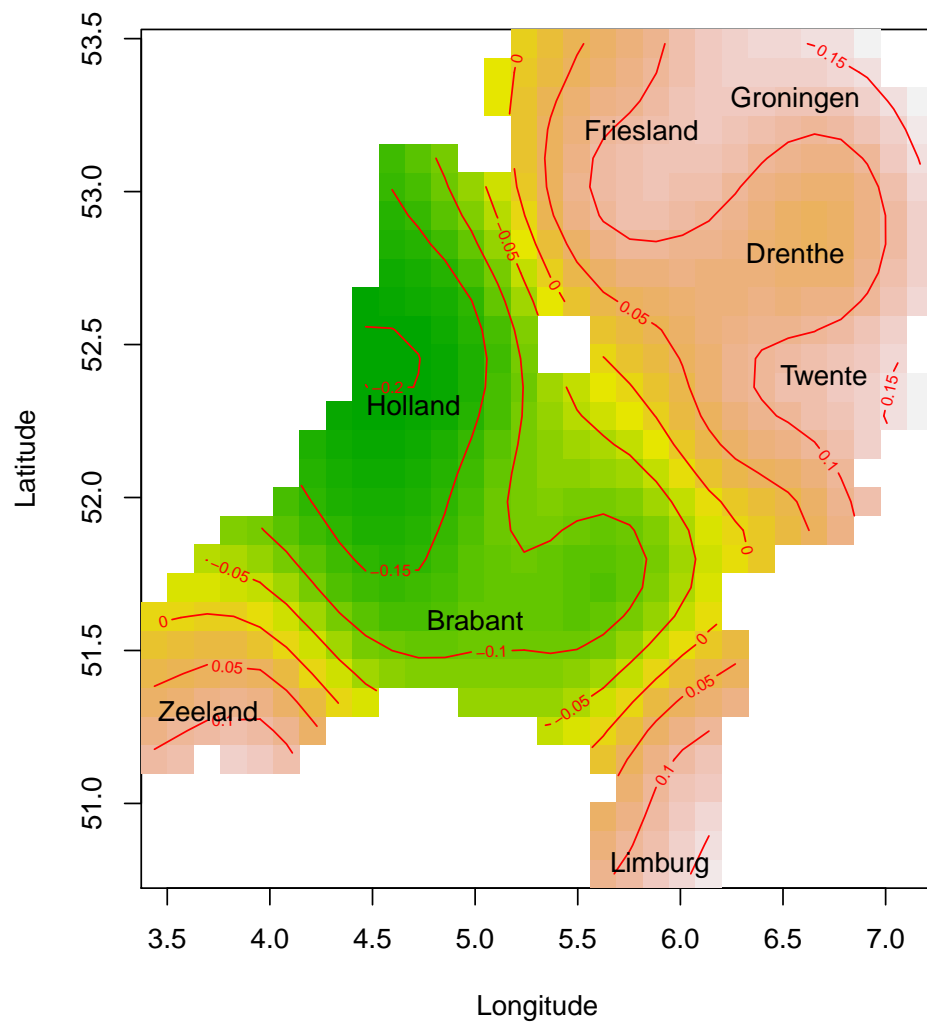
##
## Family: gaussian
## Link function: identity
##
## Formula:
## PronDistStdDutch.c ~ s(Longitude, Latitude)
##
## Parametric coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.70e-16   5.89e-04      0      1
##
## Approximate significance of smooth terms:
##              edf Ref.df    F p-value
## s(Longitude, Latitude) 28.7    29 1051 <2e-16
##
## R-sq.(adj) = 0.119   Deviance explained = 11.9%
## GCV score = 0.078321   Scale est. = 0.078311   n = 225866

dialectNL$Geo = fitted(geo)
```

#### Visualization of the generalized additive model

```
vis.gam(geo, plot.type="contour", color="terrain", too.far=0.05,
        view=c("Longitude", "Latitude"), main="")

# labels on graph
text(5.81, 53.2, "Friesland", adj=0.5)
text(6.56, 53.3, "Groningen", adj=0.5)
text(6.56, 52.8, "Drenthe", adj=0.5)
text(6.7, 52.4, "Twente", adj=0.5)
text(5.9, 50.8, "Limburg", adj=0.5)
text(3.7, 51.3, "Zeeland", adj=0.5)
text(5.0, 51.6, "Brabant", adj=0.5)
text(4.7, 52.3, "Holland", adj=0.5)
```



## Complete mixed-effects regression model

```
model <-  
  lmer(PronDistStdDutch.c ~ Geo + PopSize.log_residGeo.z +  
    PopAvgAge_residPopAvgIncome.log_Geo.z +  
    PopAvgIncome.log_residGeo.z + WordFreq.log.z +  
    WordIsNounOrAdverb + WordVCratio.log.z +  
    (1 | Word) +  
    (0 + PopSize.log_residGeo.z +  
      PopAvgAge_residPopAvgIncome.log_Geo.z +  
      PopAvgIncome.log_residGeo.z | Word) +  
    (1 + WordFreq.log.z +  
      WordIsNounOrAdverb | Location) +  
    (1 | Transcriber), dat = dialectNL  
  ) # duration: 85 minutes  
  
save(model, file='results/model.rda')
```

```
load('results/model.rda')
```

*## Note: the specification for S3 class "family" in package 'lme4' seems equivalent to one from package 'lme4.0': not turning on duplicate class definitions for this class.*  
*## Note: the specification for class "lmList" in package 'lme4' seems equivalent to one from package 'lme4.0': not turning on duplicate class definitions for this class.*  
*## Note: the specification for class "lmList.confint" in package 'lme4' seems equivalent to one from package 'lme4.0': not turning on duplicate class definitions for this class.*

```
summary(model)
```

```
## Linear mixed model fit by REML  
## Formula: PronDistStdDutch.c ~ Geo + PopSize.log_residGeo.z + PopAvgAge_residPopAvgI...  
## Data: dialectNL  
## AIC BIC logLik deviance REMLdev  
## -29416 -29179 14731 -29529 -29462  
## Random effects:  
## Groups Name Variance Std.Dev. Corr  
## Word PopSize.log_residGeo.z 3.45e-04 0.0186  
## PopAvgAge_residPopAvgIncome.log_Geo.z 7.39e-05 0.0086 -0.856  
## PopAvgIncome.log_residGeo.z 2.58e-04 0.0161 0.867 -0.749  
## Word (Intercept) 1.94e-02 0.1394  
## Location (Intercept) 3.76e-03 0.0613  
## WordFreq.log.z 2.61e-04 0.0161 -0.084  
## WordIsNounOrAdverb 2.79e-03 0.0528 -0.595 0.550  
## Transcriber (Intercept) 6.77e-04 0.0260  
## Residual 4.98e-02 0.2233  
## Number of obs: 225866, groups: Word, 559; Location, 424; Transcriber, 30  
##  
## Fixed effects:  
## Estimate Std. Error t value  
## (Intercept) -0.015252 0.010475 -1.5  
## Geo 0.968421 0.027415 35.3
```

```

## PopSize.log_residGeo.z          -0.006944    0.002632    -2.6
## PopAvgAge_residPopAvgIncome.log_Geo.z  0.004481    0.002483     1.8
## PopAvgIncome.log_residGeo.z        -0.000521    0.002621    -0.2
## WordFreq.log.z                    0.019840    0.006042     3.3
## WordIsNounOrAdverb                0.040938    0.012243     3.3
## WordVCratio.log.z                 0.062463    0.005925    10.5
##
## Correlation of Fixed Effects:
##          (Intr) Geo      PS._G. PAA_PA PAI._G WrdF.. WrINOA
## Geo          0.010
## PpSz.lg_rG.  0.015 -0.039
## PAA_PA._G.  -0.021  0.049 -0.003
## PpAvgIn._G.  0.014 -0.022  0.056 -0.070
## WrdFrq.lg.z -0.084  0.000  0.000  0.000  0.000
## WrdIsNnOrAd -0.577  0.000  0.000  0.000  0.000  0.158
## WrdVCrt.lg.  0.012  0.000  0.000  0.000  0.000 -0.035 -0.021

```

## Model comparison of fixed effects (example)

```
model.fixed1 <- lmer(PronDistStdDutch.c ~ (1 | Word) + (1 | Location)
  + (1 | Transcriber), dat = dialectNL, REML=F)

model.fixed2 <- lmer(PronDistStdDutch.c ~ Geo + (1 | Word) + (1 | Location)
  + (1 | Transcriber), dat = dialectNL, REML=F)

anova(model.fixed1,model.fixed2)

## Data: dialectNL
## Models:
## model.fixed1: PronDistStdDutch.c ~ (1 | Word) + (1 | Location) + (1 | Transcriber)
## model.fixed2: PronDistStdDutch.c ~ Geo + (1 | Word) + (1 | Location) + (1 |
## model.fixed2: Transcriber)
##           Df      AIC      BIC logLik Chisq Chi Df Pr(>Chisq)
## model.fixed1  5 -24215 -24163  12113
## model.fixed2  6 -24754 -24692  12383   541     1    <2e-16
```

## Model comparison of random effects (example)

```
model.random1 <- lmer(PronDistStdDutch.c ~ Geo + PopSize.log_residGeo.z +
  PopAvgAge_residPopAvgIncome.log_Geo.z +
  PopAvgIncome.log_residGeo.z + WordFreq.log.z +
  WordIsNounOrAdverb + WordVCratio.log.z +
  (1 | Word), dat = dialectNL)

model.random2 <- lmer(PronDistStdDutch.c ~ Geo + PopSize.log_residGeo.z +
  PopAvgAge_residPopAvgIncome.log_Geo.z +
  PopAvgIncome.log_residGeo.z + WordFreq.log.z +
  WordIsNounOrAdverb + WordVCratio.log.z +
  (1 | Word) + (1 | Location), dat = dialectNL)

anova(model.random1,model.random2)

## Data: dialectNL
## Models:
## model.random1: PronDistStdDutch.c ~ Geo + PopSize.log_residGeo.z + PopAvgAge_residP...
## model.random1: PopAvgIncome.log_residGeo.z + WordFreq.log.z + WordIsNounOrAdver...
## model.random1: WordVCratio.log.z + (1 | Word)
## model.random2: PronDistStdDutch.c ~ Geo + PopSize.log_residGeo.z + PopAvgAge_residP...
## model.random2: PopAvgIncome.log_residGeo.z + WordFreq.log.z + WordIsNounOrAdver...
## model.random2: WordVCratio.log.z + (1 | Word) + (1 | Location)
##           Df      AIC      BIC logLik Chisq Chi Df Pr(>Chisq)
## model.random1 10 -14063 -13960   7042
## model.random2 11 -24850 -24736  12436 10788     1    <2e-16
```

## By-word random slopes

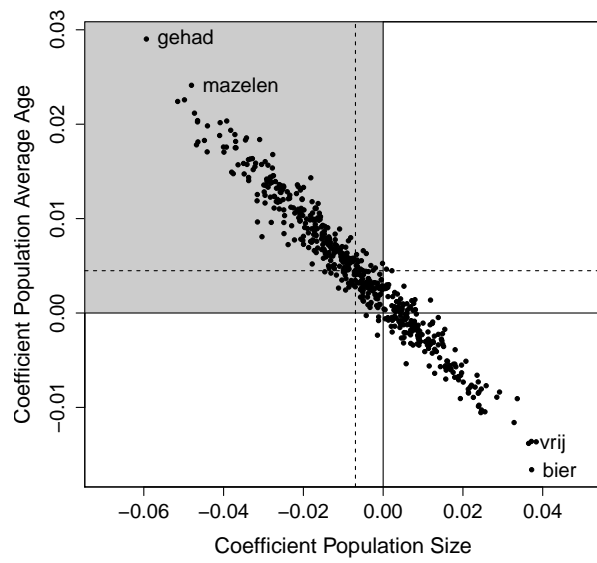
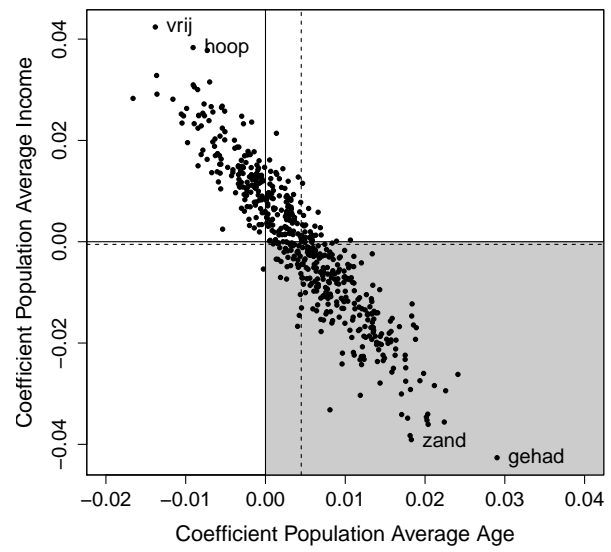
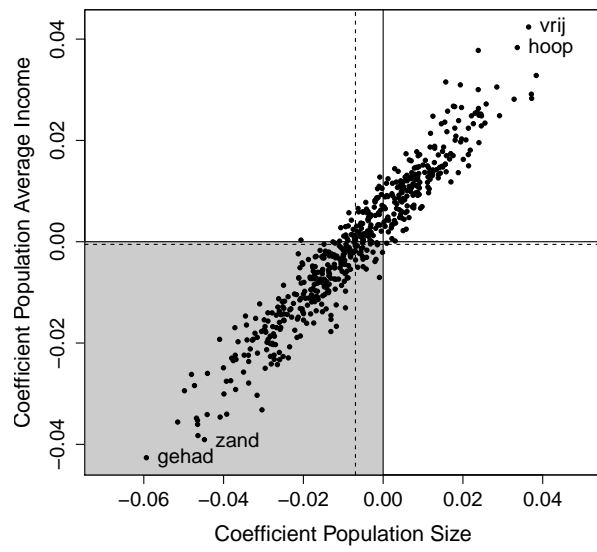
```
wordCoefs = data.frame(coef(model)$Word)
fixedEstimates = getFixefCoefs(model) # returns fixed effects coefficients

par(mfrow=c(2,2))

plotSlopes(wordCoefs,fixedEstimates,"PopSize.log_residGeo.z",
            "PopAvgIncome.log_residGeo.z","Population Size",
            "Population Average Income",-0.07,0.05)

plotSlopes(wordCoefs,fixedEstimates,"PopAvgAge_residPopAvgIncome.log_Geo.z",
            "PopAvgIncome.log_residGeo.z","Population Average Age",
            "Population Average Income",-0.02,0.04)

plotSlopes(wordCoefs,fixedEstimates,"PopSize.log_residGeo.z",
            "PopAvgAge_residPopAvgIncome.log_Geo.z","Population Size",
            "Population Average Age",-0.07,0.05)
```

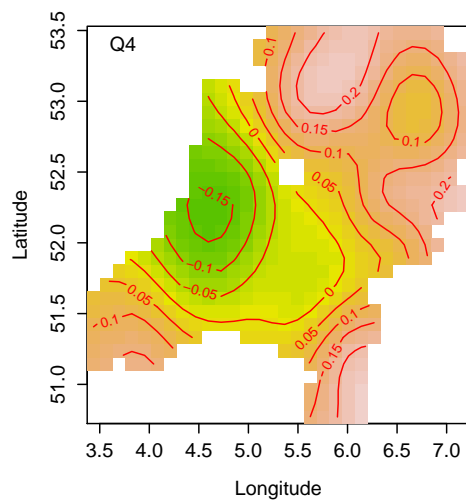
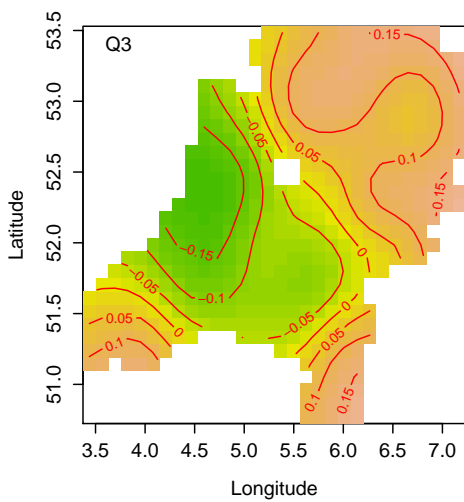
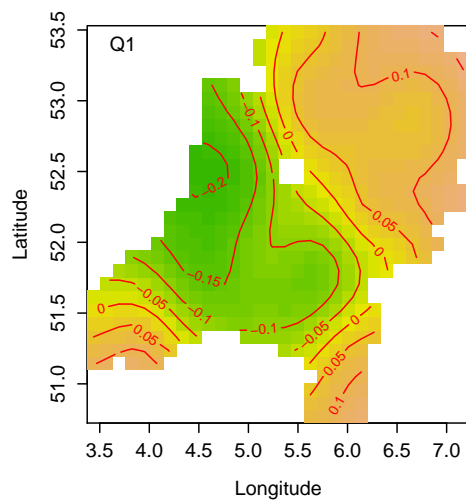
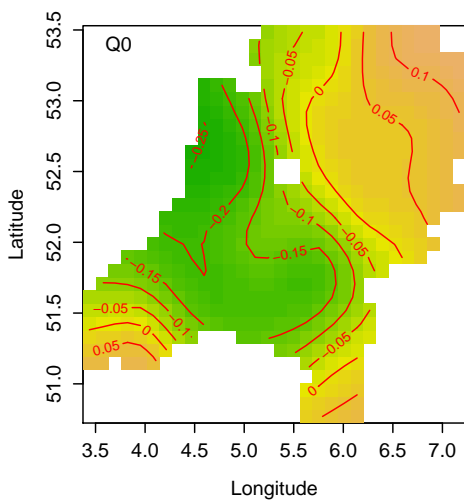
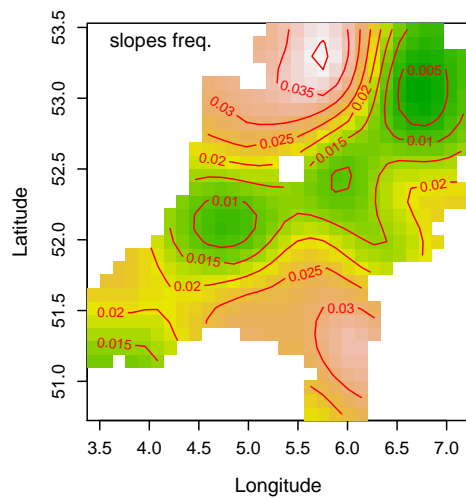
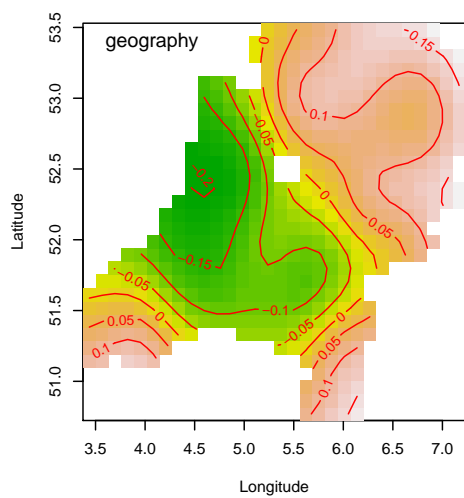


## By-location random slopes of word frequency

```
modelCoefs = coef(model)
fixedEstimates = getFixefCoefs(model)

par(mfrow=c(3,2))

vis.gam(geo, plot.type="contour", color="terrain", too.far=0.05,
        view=c("Longitude","Latitude"), main="")
text(3.6, 53.4, "geography", adj=0, cex=1.25)
plotGeoSlopes(modelCoefs, dialectNL, "WordFreq.log.z", "freq.")
plotGeoResultsQuantile(modelCoefs, fixedEstimates, dialectNL, "WordFreq.log.z", 0)
plotGeoResultsQuantile(modelCoefs, fixedEstimates, dialectNL, "WordFreq.log.z", 1)
plotGeoResultsQuantile(modelCoefs, fixedEstimates, dialectNL, "WordFreq.log.z", 3)
plotGeoResultsQuantile(modelCoefs, fixedEstimates, dialectNL, "WordFreq.log.z", 4)
```



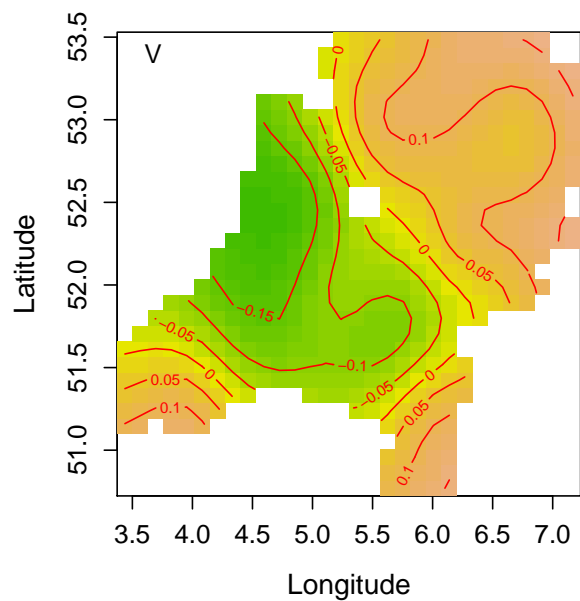
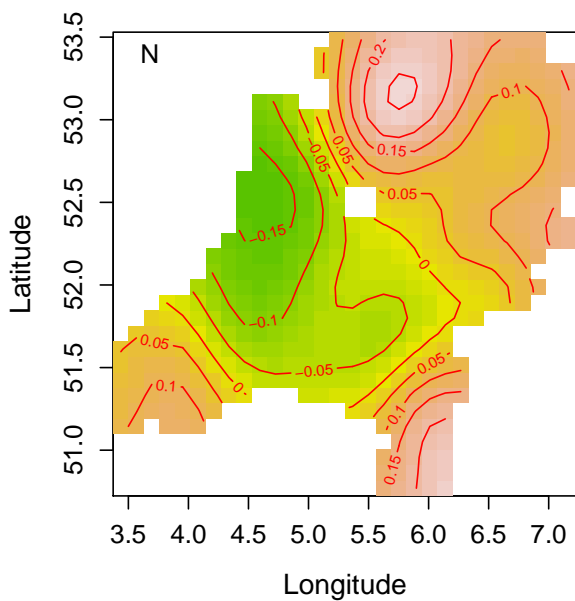
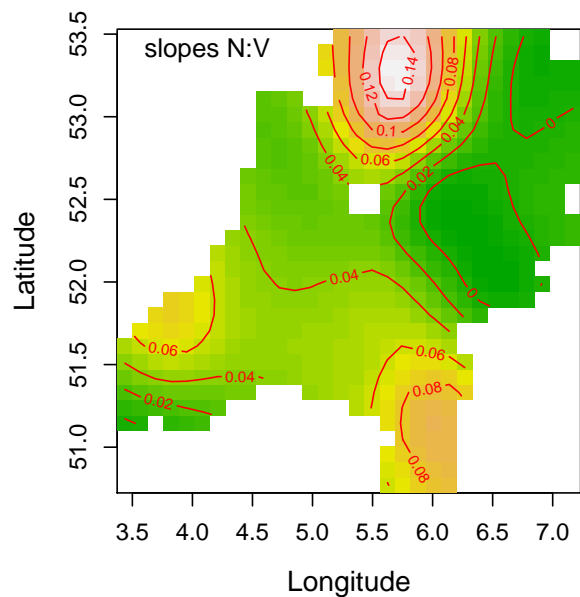
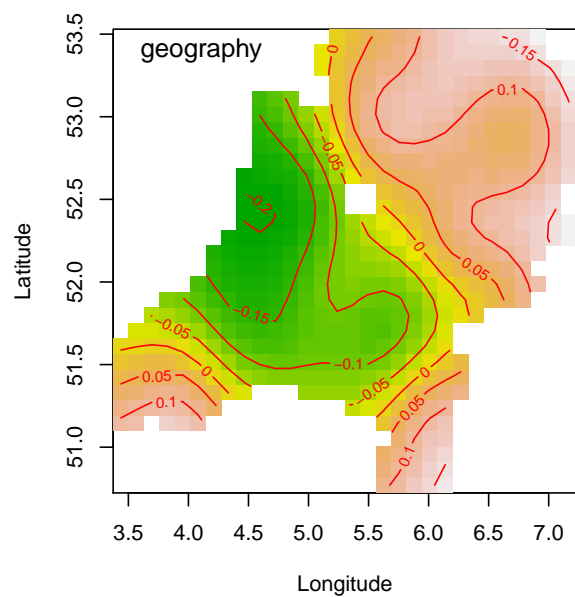
Animation illustrating the geographical pattern for increasing word frequency

## By-location random slopes of the noun-verb contrast

```
modelCoefs = coef(model)
fixedEstimates = getFixefCoefs(model)

par(mfrow=c(2,2))

vis.gam(geo, plot.type="contour", color="terrain", too.far=0.05,
        view=c("Longitude","Latitude"), main="")
text(3.6, 53.4, "geography", adj=0, cex=1.25) # add label
plotGeoSlopes(modelCoefs, dialectNL, "WordIsNounOrAdverb", "N:V")
plotGeoResultsNounVerb(modelCoefs, fixedEstimates, dialectNL, "N")
plotGeoResultsNounVerb(modelCoefs, fixedEstimates, dialectNL, "V")
```



## 4 Improved analysis

Based on the current state of the art (June 2013), we would have conducted the analysis by creating a single generalized additive mixed-effects regression model. The model below illustrates the idea (note that no model comparison procedure has been conducted, so it is likely there exist other generalized additive models which are better). Note that for this model the general effects are similar to the mixed-effects regression model, but some differences can be observed with respect to the geographical patterns. Interestingly, this analysis also shows that the frequency-based geographical pattern differs between nouns and verbs.

```
dialectNL$WordIsNounOrAdverb = as.factor(dialectNL$WordIsNounOrAdverb)

gammodel <-
  bam(PronDistStdDutch.c ~ te(Longitude, Latitude, WordFreq.log.z,
    by=WordIsNounOrAdverb, d=c(2,1)) +
    WordIsNounOrAdverb +
    PopSize.log_residGeo.z +
    PopAvgAge_residPopAvgIncome.log_Geo.z +
    PopAvgIncome.log_residGeo.z +
    WordVCRatio.log.z +
    s(Word, bs="re") +
    s(Word, PopSize.log_residGeo.z, bs="re") +
    s(Word, PopAvgAge_residPopAvgIncome.log_Geo.z, bs="re") +
    s(Word, PopAvgIncome.log_residGeo.z, bs="re") +
    s(Location, bs="re") +
    s(Transcriber, bs="re"),
    dat = dialectNL, gc.level = 2
  ) # duration: 47 minutes

gammodel.smry <- summary(gammodel) # duration: 8 minutes

save(gammodel, file='results/gammodel.rda')
save(gammodel.smry, file='results/gammodel.smry.rda')
```

```
load('results/gammodel.smry.rda')
gammodel.smry

##
## Family: gaussian
## Link function: identity
##
## Formula:
## PronDistStdDutch.c ~ te(Longitude, Latitude, WordFreq.log.z,
##   by = WordIsNounOrAdverb, d = c(2, 1)) + WordIsNounOrAdverb +
##   PopSize.log_residGeo.z + PopAvgAge_residPopAvgIncome.log_Geo.z +
##   PopAvgIncome.log_residGeo.z + WordVCRatio.log.z + s(Word,
##   bs = "re") + s(Word, PopSize.log_residGeo.z, bs = "re") +
##   s(Word, PopAvgAge_residPopAvgIncome.log_Geo.z, bs = "re") +
##   s(Word, PopAvgIncome.log_residGeo.z, bs = "re") + s(Location,
```

```

##      bs = "re") + s(Transcriber, bs = "re")
##
## Parametric coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -0.01262    0.01276   -0.99  0.3227
## WordIsNounOrAdverb1    0.03940    0.01205    3.27  0.0011
## PopSize.log_residGeo.z   -0.00778    0.00307   -2.53  0.0114
## PopAvgAge_residPopAvgIncome.log_Geo.z  0.00306    0.00297    1.03  0.3030
## PopAvgIncome.log_residGeo.z    0.00336    0.00310    1.08  0.2784
## WordVCratio.log.z        0.06248    0.00594   10.53 <2e-16
##
## Approximate significance of smooth terms:
##              edf Ref.df      F p-value
## te(Longitude,Latitude,WordFreq.log.z):WordIsNounOrAdverb0  91.9    103   21.46 < 2e-16
## te(Longitude,Latitude,WordFreq.log.z):WordIsNounOrAdverb1  91.6    103   15.68 < 2e-16
## s(Word) 548.5    554  156.06 < 2e-16
## s(Word,PopSize.log_residGeo.z) 415.0    558    3.05 < 2e-16
## s(Word,PopAvgAge_residPopAvgIncome.log_Geo.z) 214.1    558    0.64 < 2e-16
## s(Word,PopAvgIncome.log_residGeo.z) 377.5    558    2.19 < 2e-16
## s(Location) 374.7    418   39.31 2.0e-06
## s(Transcriber) 20.8     29 3041.83 3.2e-16
##
## R-sq.(adj) =  0.44   Deviance explained = 44.5%
## fREML score = -14938   Scale est. = 0.049799   n = 225866

```

## 5 Visualization of the resulting patterns

```
load('results/gammodel.rda')

zlimit=c(-0.3,0.35)

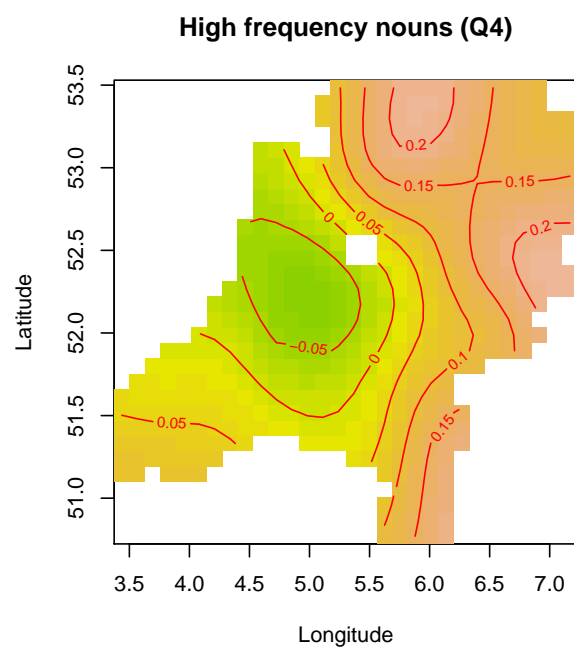
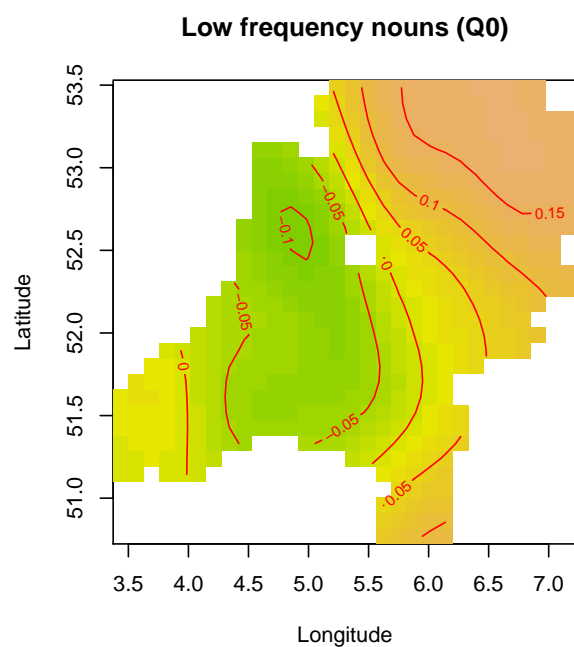
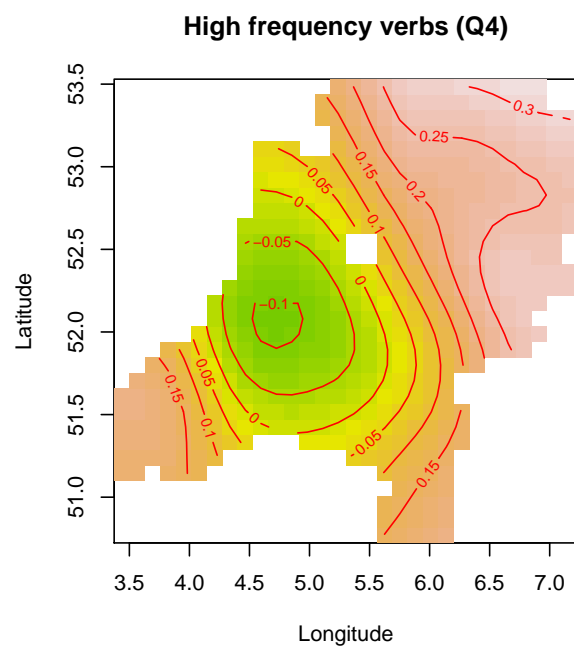
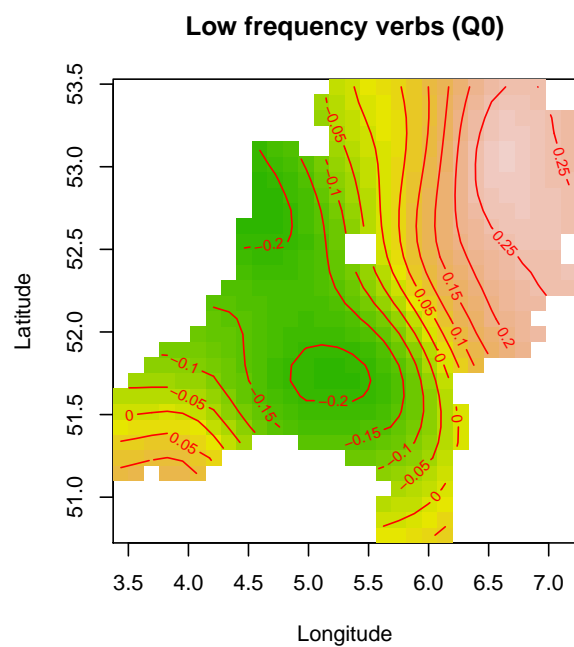
par(mfrow=c(2,2))

vis.gam(gammodel, plot.type="contour", color="terrain", too.far=0.05,
        cond=list(WordFreq.log.z=-2,WordIsNounOrAdverb="0",
                  PopSize.log_residGeo.z = 0,
                  PopAvgAge_residPopAvgIncome.log_Geo.z=0,
                  PopAvgIncome.log_residGeo.z=0, WordVCratio.log.z=0),
        view=c("Longitude","Latitude"), main="Low frequency verbs (Q0)",
        zlim=zlimit
        )

vis.gam(gammodel, plot.type="contour", color="terrain", too.far=0.05,
        cond=list(WordFreq.log.z=+2,WordIsNounOrAdverb="0",
                  PopSize.log_residGeo.z = 0,
                  PopAvgAge_residPopAvgIncome.log_Geo.z=0,
                  PopAvgIncome.log_residGeo.z=0, WordVCratio.log.z=0),
        view=c("Longitude","Latitude"), main="High frequency verbs (Q4)",
        zlim=zlimit
        )

vis.gam(gammodel, plot.type="contour", color="terrain", too.far=0.05,
        cond=list(WordFreq.log.z=-2, WordIsNounOrAdverb="1",
                  PopSize.log_residGeo.z = 0,
                  PopAvgAge_residPopAvgIncome.log_Geo.z=0,
                  PopAvgIncome.log_residGeo.z=0, WordVCratio.log.z=0),
        view=c("Longitude","Latitude"), main="Low frequency nouns (Q0)",
        zlim=zlimit
        )

vis.gam(gammodel, plot.type="contour", color="terrain", too.far=0.05,
        cond=list(WordFreq.log.z=+2, WordIsNounOrAdverb="1",
                  PopSize.log_residGeo.z = 0,
                  PopAvgAge_residPopAvgIncome.log_Geo.z=0,
                  PopAvgIncome.log_residGeo.z=0, WordVCratio.log.z=0),
        view=c("Longitude","Latitude"), main="High frequency nouns (Q4)",
        zlim=zlimit
        )
```



Animation illustrating the geographical pattern for increasing word frequency (nouns)

Animation illustrating the geographical pattern for increasing word frequency (verbs)