Analyzing phonetic data with generalized additive mixed models

Yu-Ying Chuang[1], Janice Fon[2], R. Harald Baayen[1]

1: Eberhard-Karls University of Tübingen, Germany
2: National Taiwan University, Taiwan

Abstract

This chapter provides a user's guide to analysing phonetic data with the Generalized Additive Model (GAM). We show how GAMs can be used to capture the different kinds of nonlinear effects and patterns that are ubiquitous in phonetic data. To illustrate how GAMs work, we present analyses of three datasets of Taiwan Mandarin, addressing nonlinearities in time series of experimental response times, in F0 contours, and in the geographic distribution of sociophonetic variation. By building models incrementally, we clarify the kinds of problem that arise at various stages of analysis, and show how these can be addressed within the GAM framework. In our analyses, we also show how variation between individual speakers can be accounted for.

1 Introduction

Measurements on human speech often show nonlinear patterns. Pitch contours, for example, typically do not develope linearly over time. Depending on the language and its prosodic structure, F0 contours can be quite wiggly. For example, the wiggly pitch contour of the Mandarin utterance shown in Figure 1 realizes not only the lexical tonal contour of each individual syllable, but also expresses an incredulous question. Traditionally, it is common to only consider a pre-defined subset of measurements, for instance, maximum or minimum pitch values, or formant frequencies at the mid point of a vowel. Although this is a reasonable way to clarify whether there is statistical difference at some point in time between two different curves, we miss out on more fine-grained potentially interesting patterns in the data.

Sometimes even much simpler response variables show nonlinear patterns. Figure 2 presents a scatterplot for word frequency and response times in an auditory lexical decision task (both on logarithmic scales). The data were extracted from the Massive Auditory Lexical Decision (MALD) database (Tucker et al., 2018), and restricted to monomorphemic words. Although a negative correlation is visible, it is also clear that the relationship is not strictly linear, as the downward trend tapers off for high-frequency words. The red line in Figure 2 captures the finer details of the relation between frequency and response time.

The red lines in Figures 1 and 2 were obtained with the help of a Generalized Additive Model, henceforth GAM. In this chapter, we present three examples of how GAMs can be used to probe the quantitative structure of realistic non-trivial datasets. All examples address aspects of the phonetics of Taiwan Mandarin. The first dataset contains response time data from an auditory priming experiment. With this dataset, we illustrate how to take into account subject-specific temporal adaption that takes place in the course of the experiment. For the second dataset, we measured the realizations of Mandarin high-level tones by different speaker groups in Taiwan. Here we will show how to model time-varying measurements such as pitch contours, and their interaction with factorial predictors. The third dataset addresses the ongoing merging of two sets of sibilants in

Taiwan Mandarin. Specifically, we will be looking at cross-region variation, and explain how GAMs can be used to clarify how speech varies with geography.

2 Main concepts of GAMM

In what follows, we provide a tutorial on how to analyze phonetic data with GAMs. We will therefore only introduce the basic concepts, and focus primarily on their application. For a more comprehensive explanation of GAMs, see Baayen et al. (2017) and specifically Wieling (2018), who illustrates in detail the analyses on tongue movement data recorded by electromagnetic articulography.

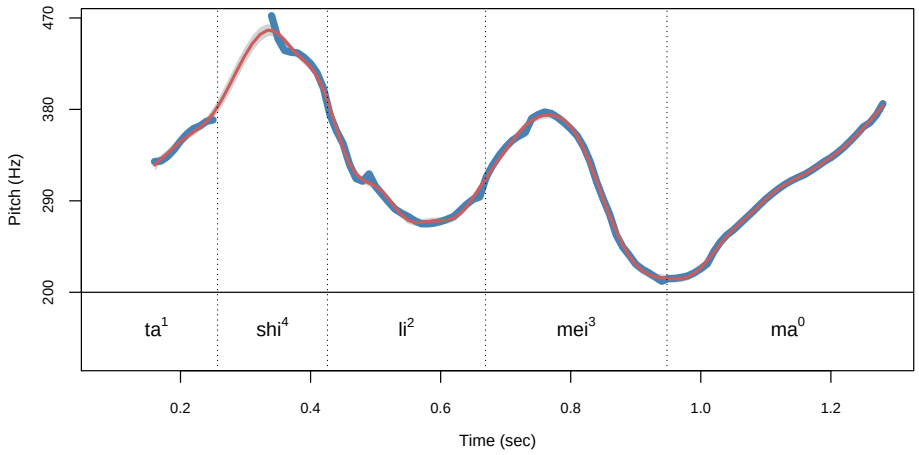

Figure 1: Pitch contour (the blue line) of the Mandarin question *ta*^{1} *shi*^{4} *Li*^{2}*mei*^{3} *ma*^{0} 'Is she Limei?'. Dotted lines indicate syllable boundaries, and the red line is the smooth curve obtained from GAM.

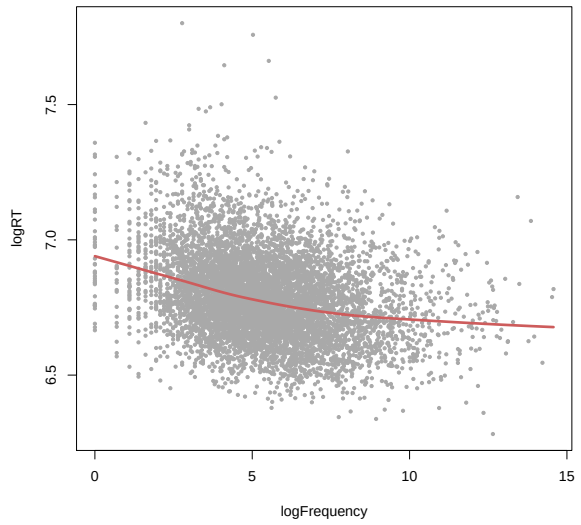

Figure 2: The effect of frequency on mean response times for monomorphemic English words in the auditory lexical decision task (data from the MALD database).

In order to model wiggly curves, GAM uses **splines**, which build on a set of nonlinear basis functions. By assigning different weights to these basis functions, a weighted sum of basis functions can approximate wiggly trends in the data. In `R`, to fit a GAM model, we use the `gam` function from the `mgcv` package (Wood, 2017). The request for a spline is achieved with the directive `s()`, which by default sets up a **thin plate regression spline**. Using the MALD data presented in Figure 2, we can fit a simple GAM model that predicts `logRT` from `logFrequency`:

```
MALDmono.gam1 = gam(logRT ~ s(logFrequency), data = MALDmono)
```

Crucially, as in a thin plate regression spline, higher-order basis functions are themselves increasingly nonlinear, approximating a more wiggly curve usually requires a larger number of basis functions. Thus, if we want to stay very faithful to the data (i.e., capturing fine details of the undulations of the curve), we will have to use a large number of basis functions. Unrestricted use of basis functions, however, may lead to overfitting. In fact, in such cases, models become unnecessarily complex. Under the assumption that simple models are to be preferred over complex ones, GAMs incorporate a mechanism that penalizes the weights of basis functions. Penalization may take the weight of a basis function completely to zero, which amounts to removing the basis function from the spline. In general, penalization not only ensures that only reasonable basis functions are retained, but also that the weights on these basis functions are not larger than necessary. A model summary is obtained as follows:

```
summary(MALDmono.gam1)
```

The summary of model `MALDmono.gam1` contains two parts.

```
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.78821    0.00178    3814   <2e-16

Approximate significance of smooth terms:
                 edf Ref.df     F p-value
s(logFrequency) 3.64  4.553 167.5  <2e-16

R-sq.(adj) =  0.0962   Deviance explained = 9.66%
GCV = 0.022797  Scale est. = 0.022782  n = 7192
```

The first part reports parametric coefficients, just as in the output summary of a liner regression model. For this model, there is only an intercept. The second part reports statistics for the smooth terms. Here `edf` denotes **effective degrees of freedom**. It gives us the sum of the proportions of the unpenalized weights that are retained in the penalized weights. Since a larger `edf` typically implies more basis functions are used, its magnitude is a rough indicater for the degree of wiggliness of the smooth. When the `edf` are equal to one or close to one, the effect is likely to be linear, as illustrated for the effect of `Duration` in model `MALDmono.gam2`.

```
MALDmono.gam2 = gam(logRT ~ s(logFrequency) + s(Duration), data = MALDmono)
```

```
                   edf Ref.df        F   p-value
s(logFrequency) 3.8256  4.767  78.1910  7.11e-75
s(Duration)     1.0000  1.000 626.8158 9.83e-135
```

This is because the first two basis functions of the spline are a horizontal line and a tilted line, the counterparts of the intercept and slope of a standard linear regression model. When `edf` are equal to one, this means that in addition to the intercept (the first basis function, which is already represented by the parametric part of the summary), only one additional basis function is needed, the basis function for the tilted line. Importantly, if an effect is linear, a GAM is able to detect this and report a linear effect.

When working with raw data, instead of aggregated data, we need to also take subject or item variability into account. For example, to allow subjects to have different intercepts, in the directive `s()` we add `bs="re"`, where `"re"` stands for random effect. For this to work, it is essential that the pertinent random effect factor is coded as such in the dataset:

```
MALDmonoRaw$Subject = as.factor(MALDmonoRaw$Subject)
MALDmonoRaw.gam1 = gam(logRT ~ s(logFrequency) + s(Duration) +
                                s(Subject, bs="re"),
                       data = MALDmonoRaw)
```

We can also change the general tilt of the regression curve for frequency by adding by-subject random slopes.

```
MALDmonoRaw.gam2 = gam(logRT ~ s(logFrequency) + s(Duration) +
                                s(Subject, bs="re") +
                                s(Subject, logFrequency, bs="re"),
                       data = MALDmonoRaw)
```

Later we will show how to model by-subject wiggly patterns (Section 3).

Often we are interested in not only the main effects of single predictors, but also in their interactions with other predictors. If the interaction of interest is between numeric and factorial predictors (e.g., an interaction between word frequency and age group), we can fit a wiggly curve for each level of the factorial predictor (See Section 4). When we are interested in an interaction between two or more numeric predictors, we can use a **tensor product smooth**. Just as splines fit a wiggly curve for the effect of a single predictor, tensor product smooths fit a wiggly surface for the joint effect of two or more predictors. This will be presented in Section 5.

In what follows, the approach that we are taking is exploratory, rather than confirmatory. One reason for this is that this allows us to introduce increasingly more intricate aspects of a model step by step. The other reason is that the complexity of the data is such that it is difficult to predict in advance how exactly predictors interact and jointly shape a regression line or surface. Due to multiple testing inherent in exploratory data analyses, we will accept an effect as potentially replicable when its associate p-value is less than 0.0001.

3 Analyzing response times to auditory stimuli

For response time data, subject variability is usually a major source of variance. To address this issue, it is common to incorporate by-subject random intercepts and slopes. These random effects are however still restricted to linearity. GAMs, on the other hand, are able to model nonlinearity for both fixed and random effects. As will be shown below, relaxing the linearity assumption enables the model to account for more variance in the data, improving model fit to a substantial extent.

There are three voiceless retroflex sibilants in Mandarin, the fricative /ʂ/, the affricate /tʂ/, and the aspirated affricate /tʂʰ/. Taiwan Mandarin, however, is characterized by reduced retroflexion. Retroflex sounds are partially or completely merging into their dental counterparts, /s/, /ts/, and

/ts^{h}/ respectively. Although Taiwan Mandarin speakers in general are not unfamiliar with deretroflexed pronunciations, it is unclear whether and how deretroflexion affects lexical processing. To address this issue, an auditory priming experiment was conducted by Chuang (2017). In this experiment, stimuli were 30 retroflex-initial bisyllabic words which were produced either with retroflex or with deretroflexed pronunciations, coded as "standard" and "variant" respectively in the column `targetType` of the dataset `ret`. There were three priming conditions: the same word with retroflex pronunciation, the same word with deretroflexed pronunciation, and a completely different word without any retroflex or dental sounds. In the `primeType` column, these conditions are referred to as "standard", "variant", and "control". To avoid the effect of speaker priming, primes and targets were produced by two different speakers. The `voice` column of `ret` specifies the speaker of the target word, "A" and "B". If the speaker of the target word is A, it also means that the speaker of the prime word is B, and vice versa. Participants were instructed to perform auditory lexical decision on target words, and their response times (in millisecond) were recorded. In the `ret` dataset, only correct responses are included.

```
head(ret,3)
```

```
  subject primeType targetType voice   RT trial      word   targetP
1      S1  standard    variant     A 1062     1    shibai 0.3597897
2      S1  standard   standard     A  834     2 shanliang 0.4546521
3      S1   control    variant     A 1472    10    zhekou 0.3102893
```

We first excluded datapoints with RTs that are shorter than 300 ms and longer than 2500 ms, and log-transformed the raw RTs to ensure that model residuals better approximate normality. We used treatment coding, with "standard" as reference level for both `targetType` and `primeType`. The first model was fitted with `targetType`, `primeType`, and `voice` as fixed-effect predictors, along with by-subject and by-word random intercepts.

```
ret.gam0 = gam(logRT ~ targetType + primeType + voice +
                       s(subject, bs="re") + s(word, bs="re"),
               data = ret)
```

```
Parametric coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)        6.822828   0.021024 324.527   <2e-16
targetTypevariant  0.075274   0.005840  12.888   <2e-16
primeTypecontrol   0.130702   0.007142  18.300   <2e-16
primeTypevariant  -0.012028   0.007131  -1.687   0.0918
voiceB             0.077066   0.005830  13.219   <2e-16

Approximate significance of smooth terms:
             edf Ref.df      F  p-value
s(subject) 45.94     47 42.990  < 2e-16
s(word)    22.60     29  3.609 2.22e-14

R-sq.(adj) =  0.516   Deviance explained = 52.9%
GCV = 0.023946  Scale est. = 0.023305  n = 2746
```

According to model summary, there is good evidence that RTs are longer if target words are produced with variant pronunciation (`targetTypevariant`). In addition, RTs are also longer when presented with a control prime (`primeTypecontrol`), since little priming can be induced by a

totally different word. RTs however becomes shorter when prime words are produced with variant pronunciation (`primeTypevariant`), but the difference is not well supported. Finally, responses are slower for target words produced by speaker B (`voiceB`). This is due to the overall slower speaking rate of speaker B.

What is also shown in the summary is that it is useful to have by-subject and by-word random intercepts included in the model. The GCV score (0.0239) is an indication of the goodness of model fit (the lower the score, the better the model fit). However, we can improve the model by taking further details of subject variability into account. Figure 3 displays RTs as a function of trial number in the experiment, for each subject separately. We can see that in addition to differences in intercepts, how RTs develop in the course of the experiment also differs substantially across subjects. S48, for example, becomes faster as the experiment procedes. By contrast, S19 has the tendency to become slower. It is thus also useful to include by-subject random slopes for trials. To avoid that `trial` dominates prediction accuracy, we first scaled it and created a new numeric variable `trialScaled`.

```
ret.gam0a = gam(logRT ~ targetType + primeType + voice +
                        s(subject, bs="re") + s(subject, trialScaled, bs="re") +
                        s(word, bs="re"),
                data = ret)
```

The GCV score of this model (0.0221) is lower than that of the previous model (`ret.gam0`), indicating improved model fit. However, Figure 3 also shows that RT development is notably not linear. For instance, S39 exhibits a convex pattern, whereas a concave pattern is found for S6. We therefore need **factor smooths**. What a factor smooth does is to fit a wiggly curve for each individual subject, with the same calibration for penalization across all subjects. A factor smooth is the nonlinear equivalent to the combination of random intercepts and random slopes in the linear mixed model, as it also implements shrinkage. In model `ret.gam1`, a factor smooth was requested by specifying `bs="fs"`. The directive `m=1` enables shrinkage for the basis function for the tilted line. Thus, if there are no linear trends for the individual subjects, then the coefficients of the linear basis function can be shrunk down all the way to zero. In this case, a plot such as Figure 4 will show a set of horizontal lines, each with its own intercept. In other words, the factor smooth has become functionally equivalent to a random effect with by-subject random intercepts.

```
ret.gam1 = gam(logRT ~ targetType + primeType + voice +
                        s(trialScaled, subject, bs="fs", m=1) +
                        s(word, bs="re"),
               data = ret)
```

The lower GCV score (0.0215) indicates that model fit again improves. The fitted smooths for all subjects are presented in Figure 4.

Next we asked whether there are interactions among the factorial predictors. Of specific interest is whether RTs to variant target words are differentially influenced by different primes and different voices. We therefore built our next model including an interaction between `targetType` and `primeType`, and in addition an interaction between `targetType` and `voice`.

```
ret.gam2 = gam(logRT ~ targetType * primeType + targetType * voice +
                        s(trialScaled, subject, bs="fs", m=1) + s(word, bs="re"),
               data = ret)
```

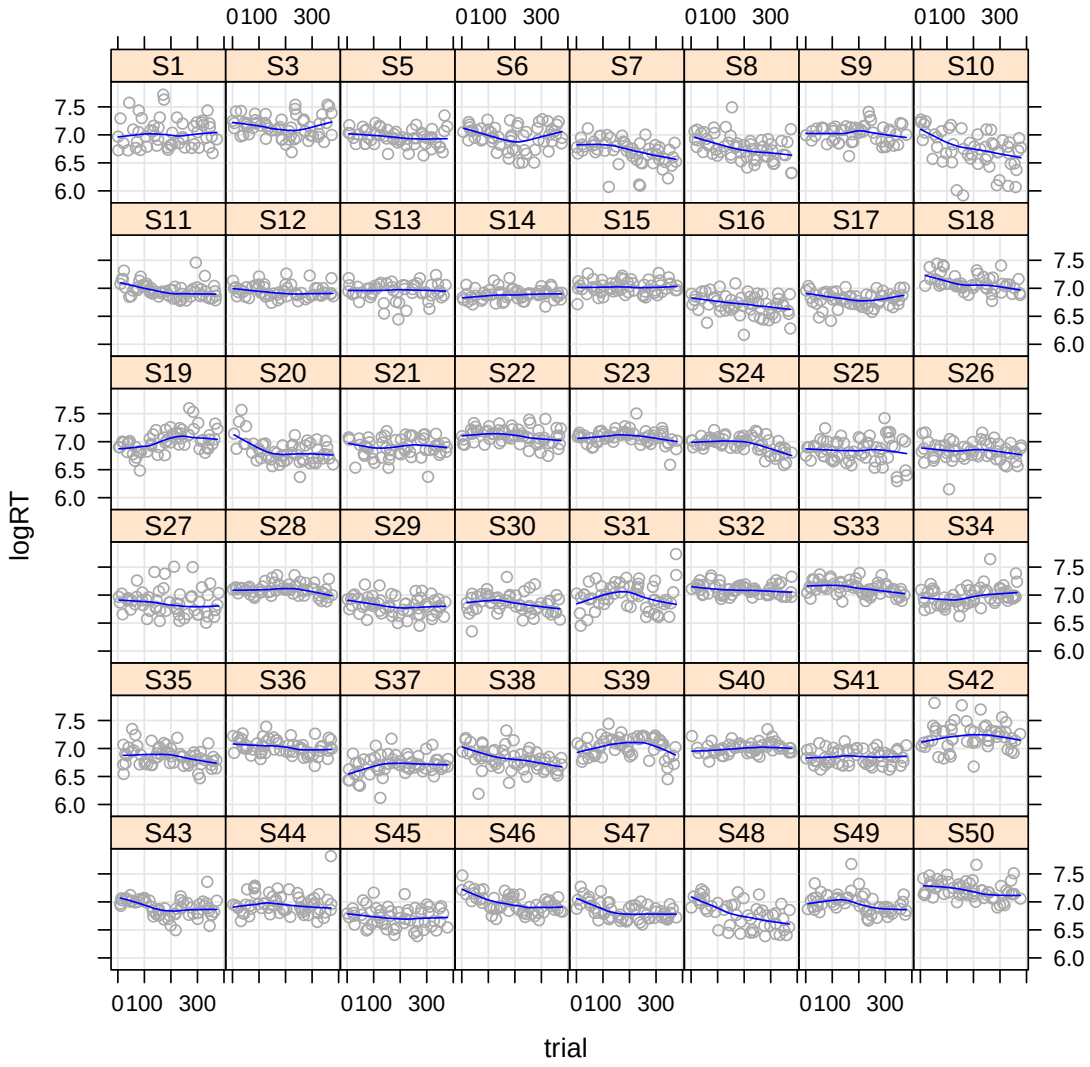

Figure 3: RTs across trials for each subject.

```
Parametric coefficients:
                                  Estimate Std. Error t value Pr(>|t|)
(Intercept)                       6.838974   0.020115 339.990  < 2e-16
targetTypevariant                 0.054803   0.010727   5.109 3.49e-07
primeTypecontrol                  0.127271   0.009180  13.864  < 2e-16
primeTypevariant                 -0.030184   0.009121  -3.309 0.000948
voiceB                            0.061979   0.008382   7.394 1.92e-13
targetTypevariant:primeTypecontrol  0.005855   0.013186   0.444 0.657071
targetTypevariant:primeTypevariant  0.037840   0.013173   2.873 0.004104
targetTypevariant:voiceB          0.015086   0.010781   1.399 0.161859
```

Model summary suggests that there is some hint of evidence that an upward adjustment of the intercept is needed when both the target and prime words are presented with variant pronunciations. The interaction between `targetType` and `voice`, on the other hand, does not seem to be necessary.

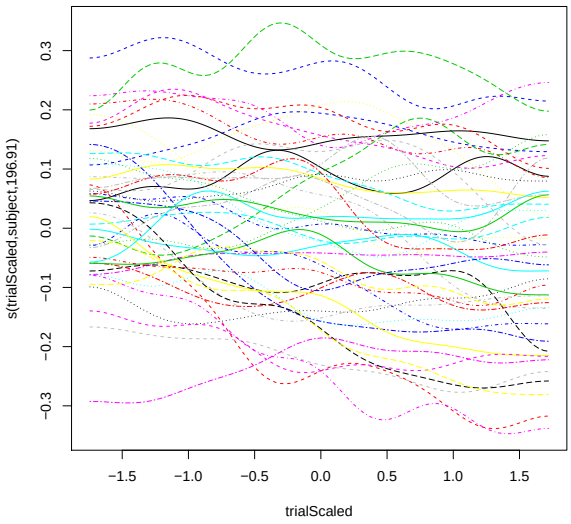

Figure 4: By-subject factor smooths for model `ret.gam1`.

Given that the GCV score (0.02149) remains practically the same as that of the model without interactions (`ret.gam1`), we should stay wary about the interpretation of these interaction effects.

In the next model we investigated the effect of a numeric predictor, `targetP`. This measure attempts to estimate the probability that a retroflex pronunciation activates the intended semantic field. When retroflexes are deretroflexed (e.g., /ʂən/ → [sən]), it is inevitable that more lexical candidates will emerge, because now in addition to words beginning with /ʂən/, words beginning with /sən/ will be compatible with the acoustic input as well. Thus, the probability of obtaining the correct semantics is usually smaller for variant pronunciations than for standard pronunciations.

To explore whether the effect of `targetP` might be nonlinear, we included it in a smooth term. Since the interaction between `targetType` and `voice` was not significant, we left it out in the model shown here.

```
ret.gam3 = gam(logRT ~ targetType * primeType + voice +
                       s(targetP) + s(trialScaled, subject, bs="fs", m=1) +
                       s(word, bs="re"),
               data = ret)
```

```
Approximate significance of smooth terms:
                          edf  Ref.df     F  p-value
s(targetP)              5.078   5.816 2.745   0.0183
s(trialScaled,subject) 197.611 431.000 6.797  < 2e-16
s(word)                 21.736  29.000 3.581 2.53e-16
```

The summary provides some minimum support for the effect of `targetP`, but in an exploratory analysis such as presented here, it is questionable whether this effect is replicable. To visualize it, we used the `plot` function, as shown below. The directive `select=1` requests the first effect in the summary table for smooth terms to be plotted. To zoom in on the effect, we set the range of the y-axis between -0.1 and 0.1. In addition, we added shading to the area of the confidence interval, and also a horizontal line at zero.

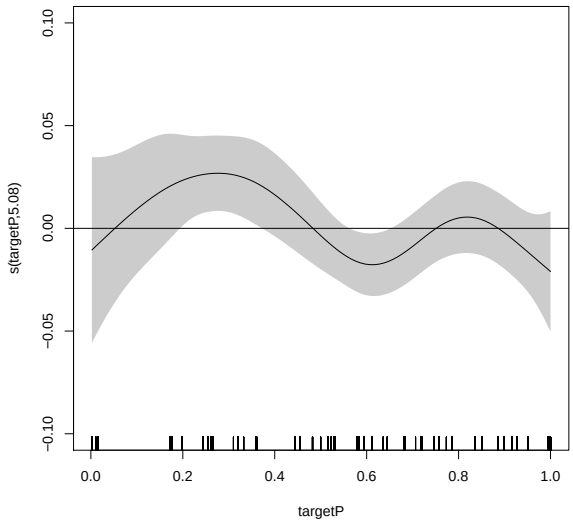

Figure 5: The effect of `targetP` from `ret.gam3`.

```
plot(ret.gam3, select=1, ylim=c(-0.1,0.1), shade=TRUE)
abline(h=0)
```

The `plot` function visualizes the partial effect of `targetP`, i.e., the population intercept and the effects of other predictors are not included. This is why the curve is centered around the y-axis. As shown in Figure 5, there is a tendency for RTs to become shorter as the probability of obtaining the intended meaning increases. As a next step, we further examined whether `targetP` interacts with `voice`. We added the directive `by=voice` to the smooth term, requesting two separate smooths, one for each voice.

```
ret.gam4 = gam(logRT ~ targetType * primeType + voice +
                       s(targetP, by=voice) +
                       s(trialScaled, subject, bs="fs", m=1) + s(word, bs="re"),
               data = ret)
```

```
                          edf  Ref.df     F  p-value
s(targetP):voiceA       4.327   5.100 1.404  0.22028
s(targetP):voiceB       7.545   8.313 3.010  0.00212
```

Apparently, the effect of `targetP` is only found for voice B, but not voice A. This explains why the effect of `targetP` is not well supported when both voices are considered together in model `ret.gam3`. The visualization of the two effects is presented in Figure 6. In the left panel, the confidence interval always includes zero, indicating that there is no significant effect anywere. By contrast, in the right panel, there is only a small portion of the curve whose confidence interval includes zero, suggesting that the effect is supported.

Although now the downward trend for voice B is clearer, the curve is still uninterpretably wiggly. In fact, the high degree of wiggliness is likely to be a technical artifact. The values of targetP are sparsely distributed, as can be seen from the rugs on the x-axis. Given that there are only 30 target

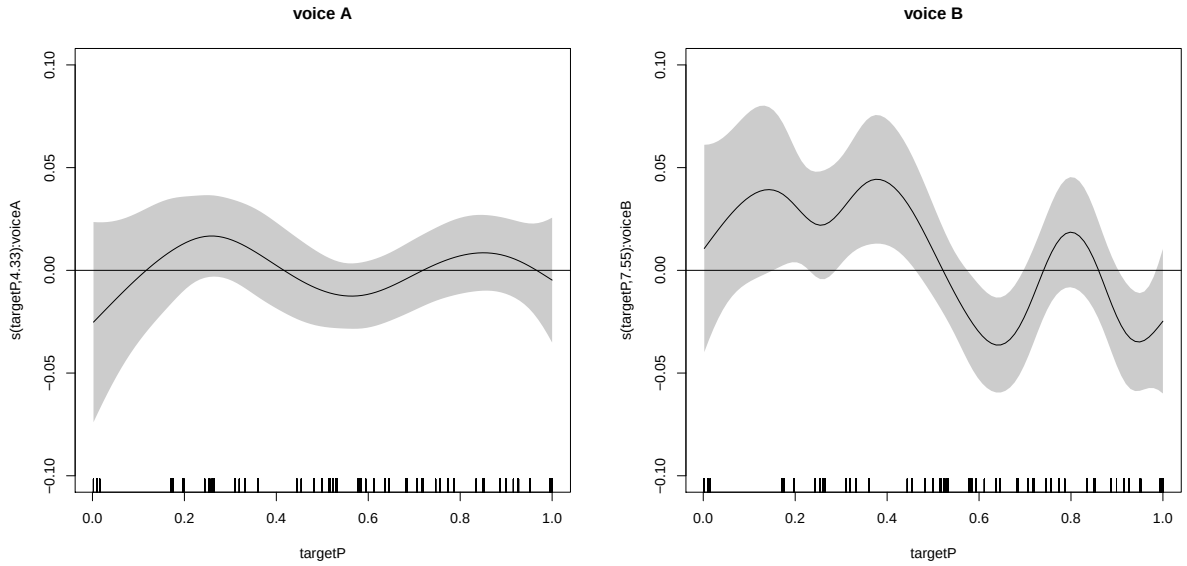

Figure 6: The effects of `targetP` for voice A (left) and voice B (right) from `ret.gam4`.

words, and that each word comes with two pronunciations (standard *vs.* variant), there are at most 60 unique values for targetP. When there are many datapoints with the same targetP value, as is the case in the present experiment in which 48 subjects responded to each target token, the GAM is presented with substantial evidence about where the mean should be for each stimulus. As a consequence, it does its best to bring the predictions as close as possible to the observed values. This then leads to overfitting. To remedy this, we brought down the number of basis functions used to fit the curves. The default number of basis functions is ten. In the following model, we set this number to four by specifying `k=4`.

```
ret.gam5 = gam(logRT ~ targetType * primeType + voice +
                       s(targetP, by=voice, k=4) +
                       s(trialScaled, subject, bs="fs", m=1) + s(word, bs="re"),
               data = ret)
```

```
Approximate significance of smooth terms:
                        edf Ref.df     F  p-value
s(targetP):voiceA      1.00      1 0.031  0.86110
s(targetP):voiceB      1.00      1 8.658  0.00328
```

Now the effects of targetP for both voices are practically linear, as both `edfs` are equal to 1. The linearity of the effects can also be seen in Figure 7. As before, there is no effect for voice A. The right panel suggests that for voice B, there might actually be a linear effect of `targetP`. However, without further replication, no firm conclusions can be drawn.

In summary, the analysis presented in this section illustrated how to use factor smooths to model the nonlinear random effect for individual subjects. In addition, we note that we should be cautious about smooths that are excessively wiggly, as they might be artifacts resulting from sparsely represented numerical predictors in repeated measurement experimental designs. This issue can be addressed by adjusting the number of basis functions.

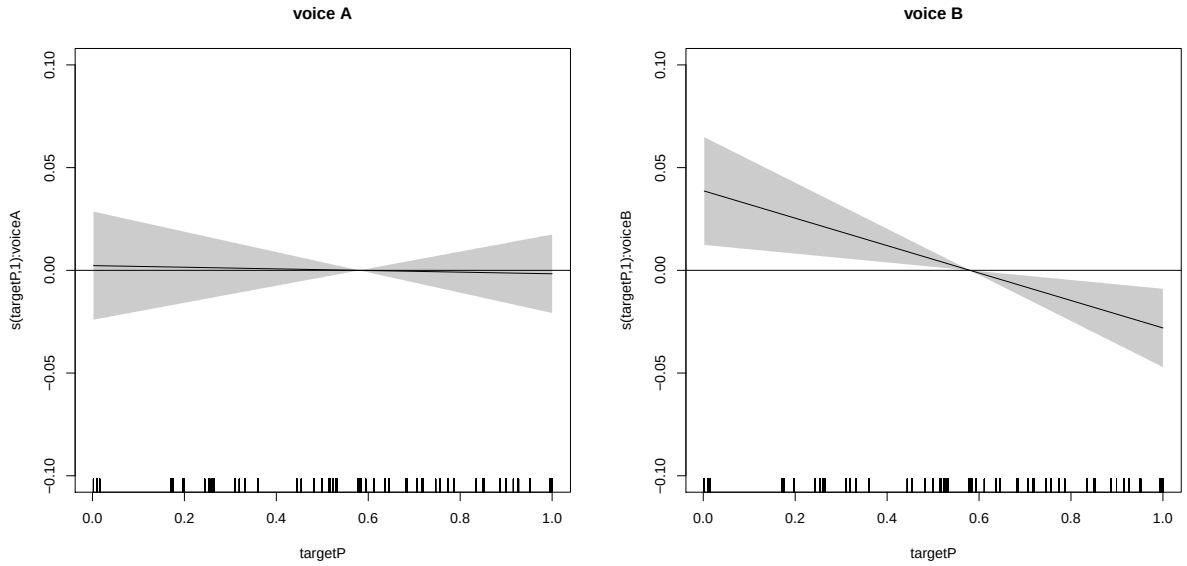

Figure 7: The effects of `targetP` for voice A (left) and voice B (right) from `ret.gam5`.

4 Time series of acoustic measurements: F0 contours

A lot of phonetic studies involve the measurement of F0. Statistical analyses, however, seldom take the entire contour into consideration. In this section we will illustrate how to model time-varying measurements such as F0 contours with a GAM. We will also show how to locate where the contours of two contrast levels are significantly different. Finally we will introduce ways to compare and critize models.

Among the four lexical tones in Mandarin, Tone 1, henceforth T1, is canoncially described as a high-level tone. To investigate whether T1 realizations in Taiwan Mandarin exhibit any dialectal variation, 25 native speakers were recorded (Fon, 2007). Thirteen speakers (6 males, 7 females) were from Taipei, and 12 (6 males, 6 females) were from Taichung (specified in the column `location` of the dataset `tone`). These are two metropolitan cities in Taiwan, located in northern and central Taiwan, respectively. Stimuli were 24 bisyllabic words, and the target T1 syllables occur in either the first (P1) or the second (P2) position of the bisyllabic words. The adjacent preceding or following syllable carried one of the four lexical tones (T1, T2, T3, T4). In total, the manipulation of position and adjacent tone gave rise to eight different tonal contexts (specified in the column `context`). By way of example, the word *bing*^{3}*gan*^{1} 'cookie' has the target syllable *gan*^{1} in P2, and is preceded by a T3 syllable, and hence the context coding is "P2.T3". To capture the pitch contour of T1, the F0 values at ten equally-spaced time points of each target syllable were measured. The data from this experiment are provided in the `tone` dataset.

```
head(tone, 3)
```

```
  subject sex location   word tarSyll position adTone context time    pitch
1     CWQ   M TAICHUNG bandai     ban       P1     T4   P1.T4    1 161.8738
2     CWQ   M TAICHUNG bandai     ban       P1     T4   P1.T4    2 159.1713
3     CWQ   M TAICHUNG bandai     ban       P1     T4   P1.T4    3 156.3541
```

Figure 8 presents T1 realizations in the eight different contexts by the four speaker groups (broken down by `sex` and `location`). In general, females have higher pitch values than males,

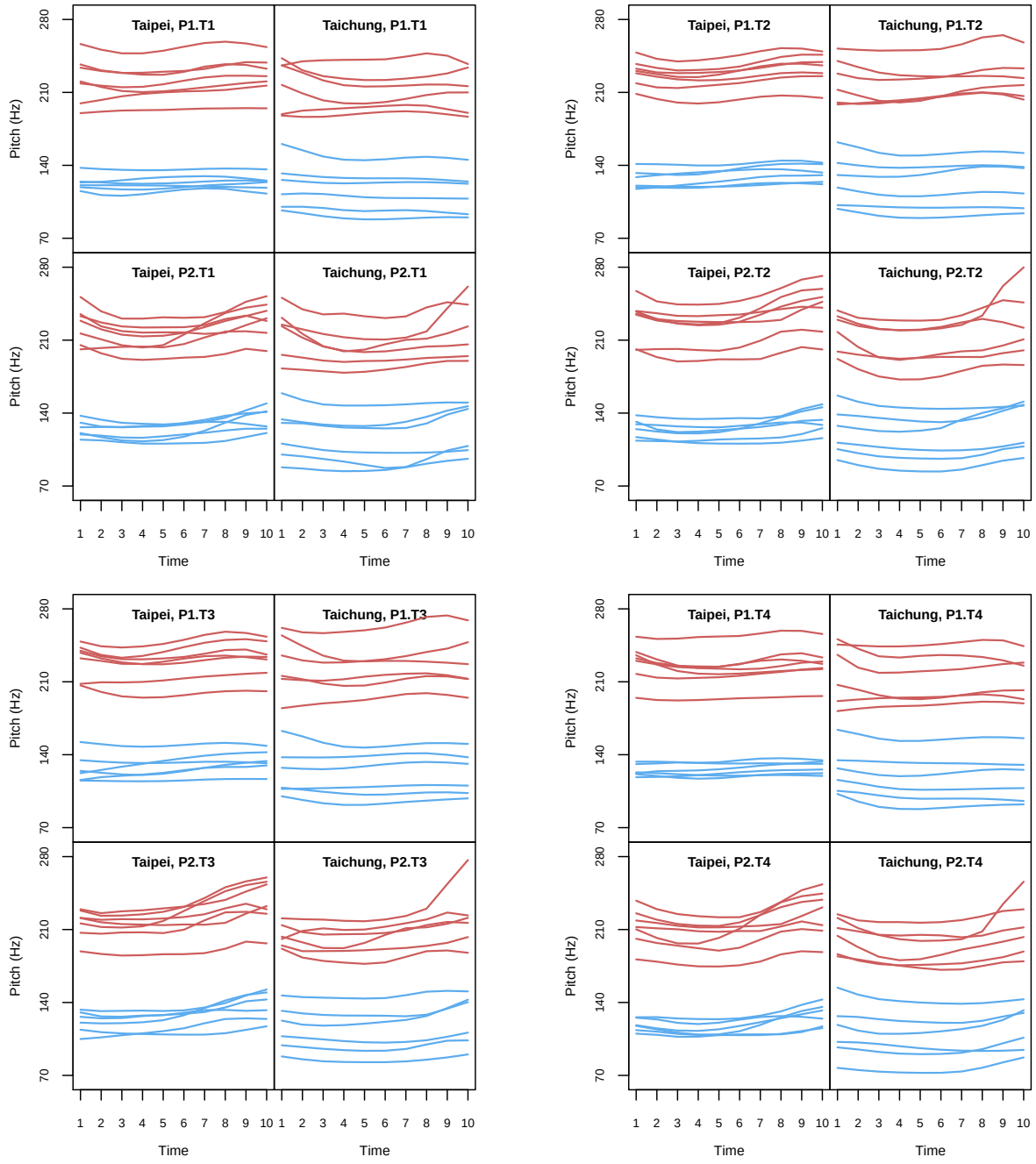

Figure 8: T1 contours of different contexts for individual male (blue) and female (red) subjects.

as expected. Interestingly, T1 is not really a "level" tone as canonically described. Instead, it is featured by a final rise, as is clearly visible especially for females and for word-final positions (P2).

Before fitting models, we centered `time`. `Time` is a numeric variable ranging from one to ten in normalized time. Centering was implemented because the model intercept, the y-coordinate where the regression line hits the y-axis, is not interpretable, since pitch at time 0 does not make sense. We centered `time` so that the intercept now will represent the mean pitch in the middle of the

target syllables. We fitted our first model with `sex`, `location`, and `context` as fixed-effect factors, along with a smooth for the variable of scaled time (`timeScaled`). In addition, we also requested a by-subject factor smooth for `timeScaled` (i.e., a wiggly line for each subject) and random intercepts for `word`. Also note that here we used `bam` instead of `gam`. This is because when working with a large dataset, `bam` works more efficiently at minimum cost of accuracy. In addition, with the directive `discrete` set to `TRUE`, covariates are binned in a mathematically principled way to enable faster estimation of the model's coefficients.

```
tone$timeScaled = as.numeric(scale(tone$time, center=TRUE, scale=FALSE))
tone.gam0 = bam(pitch ~ sex + location + context + s(timeScaled) +
                        s(timeScaled, subject, bs="fs", m=1) +
                        s(word, bs="re"),
                data=tone, discrete=TRUE)
```

```
Parametric coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)       123.505      6.337  19.489  < 2e-16
sexF               98.017      7.156  13.698  < 2e-16
locationTAICHUNG   -8.739      7.156  -1.221   0.2220
contextP2.T1       -1.196      1.163  -1.028   0.3040
contextP1.T2        6.371      1.163   5.476 4.52e-08
contextP2.T2        2.295      1.163   1.973   0.0486
contextP1.T3        7.601      1.163   6.533 6.99e-11
contextP2.T3       -1.943      1.163  -1.670   0.0950
contextP1.T4        2.108      1.163   1.812   0.0701
contextP2.T4       -9.670      1.163  -8.312  < 2e-16

Approximate significance of smooth terms:
                          edf  Ref.df      F  p-value
s(timeScaled)           5.677   6.567  27.66  < 2e-16
s(timeScaled,subject) 119.356 225.000 150.77        1
s(word)                13.698  16.000   5.95 3.89e-16

R-sq.(adj) =  0.973   Deviance explained = 97.4%
fREML =  21572  Scale est. = 73.039    n = 6000
```

According to the model summary, pitches have to be adjusted upwards for females, unsurprisingly. While the effect of `location` is not supported, `context` shows up with several well-supported contrasts. With respect to the smooth terms, the overall contour of T1, as shown in Figure 9, indeed has a dipping curvature. The by-subject factor smooth, on the other hand, is not significant, indicating that inter-subject variability is not substantial.

There is, however, one serious problem for this model. When we apply the autocorrelation function to the residuals of the model,

```
acf(resid(tone.gam0))
```

we obtain the plot presented in the left panel of Figure 10. It suggests that there is still structure remaining in the residuals, violating the crucial modeling assumption that the residuals are independent of each other. That is, residuals at time t are correlated with residuals at preceding time steps *t_lag*. At shorter lags, autocorrelations are strongest. This autocorrelation is inevitable, since

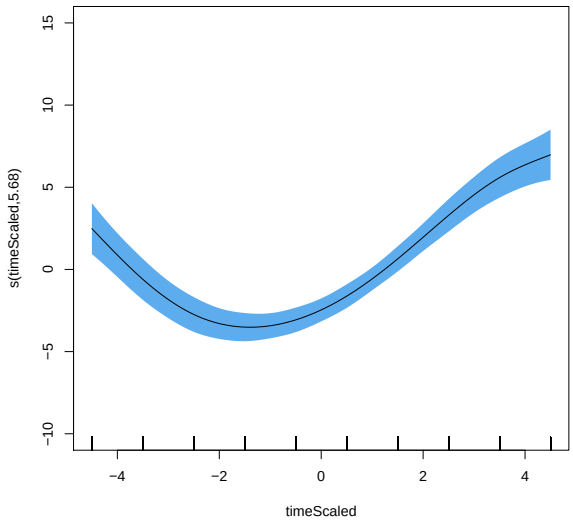

Figure 9: Partial effect of `timeScaled` from model `tone.gam0`.

given physical constraints, the vibration of the vocal folds at time t cannot be completely independent of that at time $t-1$. One strategy to deal with autocorrelated residuals is to incorporate an AR(1) process in the model. This correction process assumes that the error at time t is the sum of a proportion ρ of the preceding error at time $t-1$ and Gaussian noise. Adding an AR(1) component to our model requires that we add a further variable to the dataframe. The required variable specifies the beginning of each production (i.e., time 1) with the logical value `TRUE`, and all the others with the value `FALSE`. We now fit a new model with the AR(1) process incorporated, setting ρ to 0.8, the approximate value at lag 1 in the left panel of Figure 10.

```
tone$AR.start = FALSE
tone$AR.start[tone$time==1] = TRUE
tone.gam1 = bam(pitch ~ sex + location + context + s(timeScaled) +
                        s(timeScaled, subject, bs="fs", m=1) +
                        s(word, bs="re"),
                data=tone, discrete=TRUE, AR.start=tone$AR.start, rho=0.8)
```

To inspect the residuals, this time we used the `resid_gam` function from **itsadug** package (van Rij et al., 2017). This function discounts the part of residuals that has been taken care of by the AR(1) process.

```
acf(resid_gam(tone.gam1))
```

As shown in the right panel of Figure 10, autocorrelation in the errors has been successfully eliminated.

To further examine the effect of `context`, we fitted a new model and requested a separate smooth for each context. In order to compare the model fit of this model with that of the previous model, we used the method of maximum likelihood estimation with the directive `method = "ML"`. It is important that we cannot use the method of (fast) restricted maximum likelihood estimation, i.e., (f)REML, if the models to be compared have different fixed-effect structures.

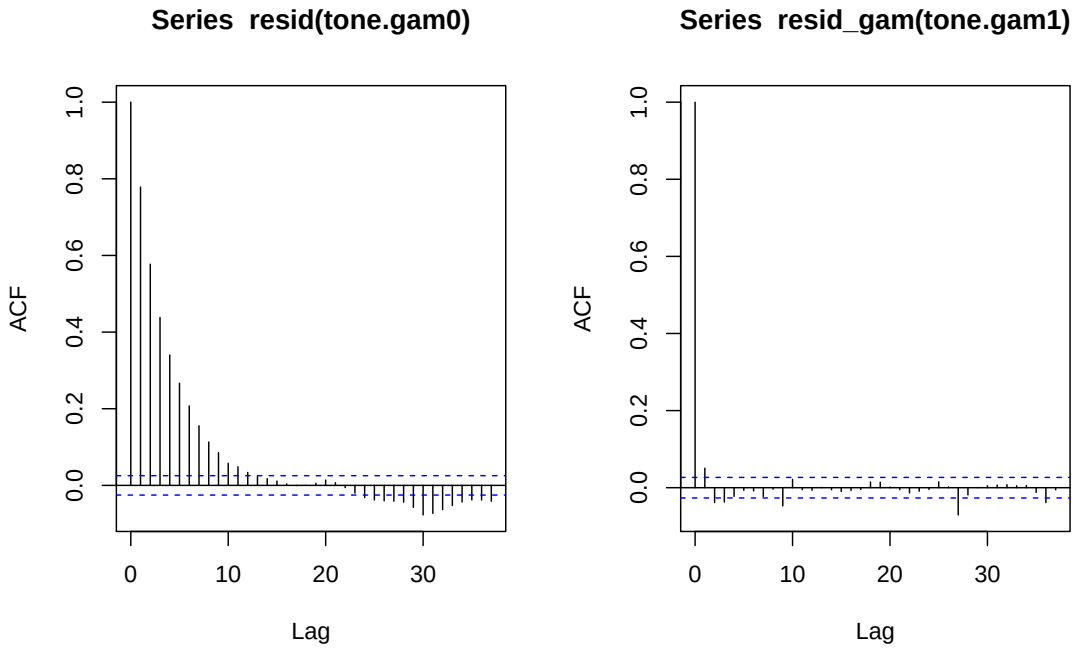

Figure 10: Autocorrelation functions fitted to model `tone.gam0` (left) and `tone.gam1` (right). Lag indicates the number of time steps preceding a given time t.

```
tone.gam2 = bam(pitch ~ sex + location + context + s(timeScaled, by=context) +
                        s(timeScaled, subject, bs="fs", m=1) +
                        s(word, bs="re"),
                data=tone, method="ML", AR.start=tone$AR.start, rho=0.8)
```

After also refitting the previous model with maximum likelihood estimation, we can now compare the model fits.

```
compareML(tone.gam1, tone.gam2)
```

```
Chi-square test of ML scores
-----
      Model    Score Edf Difference     Df  p.value Sig.
1 tone.gam1 16925.54  15
2 tone.gam2 16578.79  29    346.752 14.000  < 2e-16  ***
```

Model fit has improved substantially by allowing each combination of position and lexical tone to have its own smooth (as indicated by lower ML scores). Figure 11 clarifies that T1 is realized very differently when in `P1` compared to when in `P2`. Furthermore, tonal contours are also clearly depending on the tones of preceding and following syllables.

Although T1 looks very different for several of the eight contexts, it can however not be straightforwardly inferred from the model where pairs of contours are significantly different. For example, in the contexts of "P2.T2" and "P2.T3" (lower mid panels, Figure 11), the T1 contour in the "P2.T3" context appears to start lower than that in "P2.T2", suggesting a clear influence from preceding

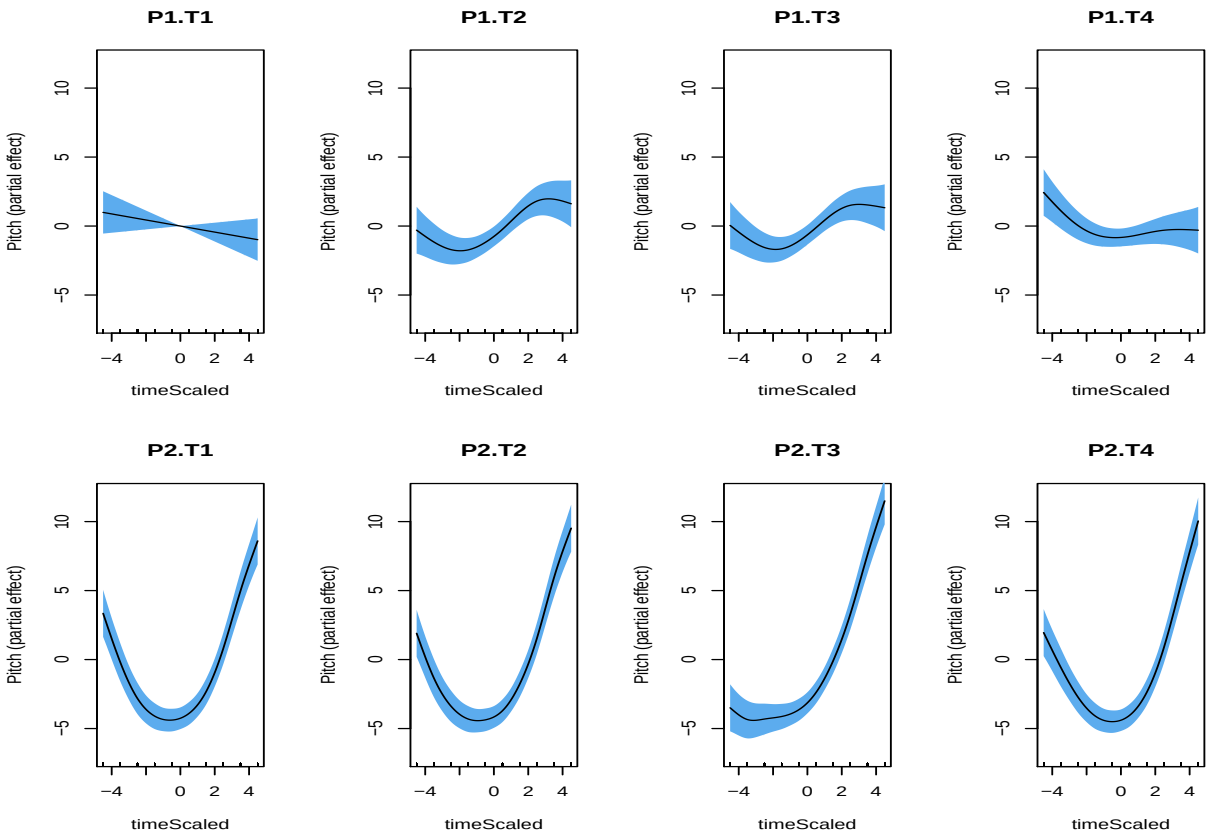

Figure 11: Partial effects of pitch contours for different contexts.

tones. However, we do not know whether this difference is statistically supported. To address this issue, we can build a difference curve.

By way of example, to zoom in on this particular contrast, we first subsetted the data to include only the pertinent datapoints. We then transformed the variable `context` into a numeric variable, with the reference level "P2.T2" coded as 0 and the contrast level "P2.T3" coded as 1. A new model was fitted, in which we included not only a smooth for the general effect of `timeScaled`, but also a smooth for the difference between the reference and the contrast levels. What this second smooth term does is to provide a corrective curve for the datapoints that fall under the contrast level. In this way, we obtain a difference curve for the two levels.

```
toneP2 = droplevels(tone[tone$context=="P2.T2"|tone$context=="P2.T3", ])
toneP2$contextNum = as.numeric(toneP2$context)-1
toneP2.gam = bam(pitch ~ sex + location +
                         s(timeScaled) + s(timeScaled, by=contextNum) +
                         s(timeScaled, subject, bs="fs", m=1) + s(word, bs="re"),
                 data=toneP2, discrete=TRUE, AR.start=toneP2$AR.start, rho=0.8)
```

The difference curve for the present example is shown in the left panel of Figure 12. T1 contours are always lower when following T3 than when following T2: the entire curve lies under the x-axis. To better understand the difference curve, we plotted the fitted values for the two contexts in the right panel of Figure 12. The difference is greater at the beginning and gradually attenuates towards the end, which also indicates that the difference decreases over time. Importantly, the confidence

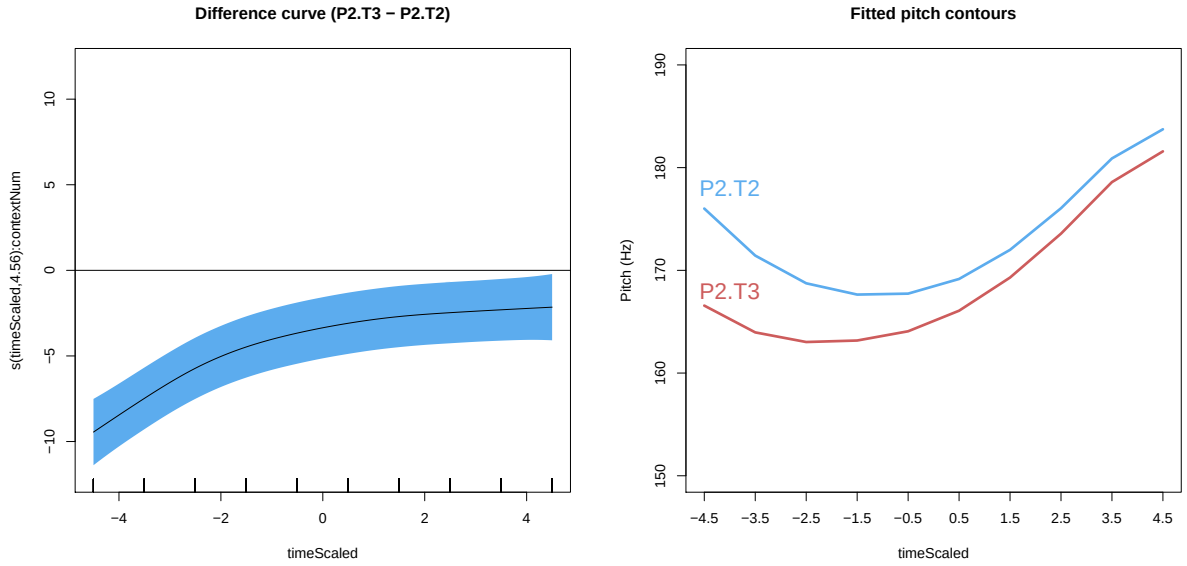

Figure 12: Difference curve between "P2.T2" and "P2.T3" contexts (left) and the predicted pitch values for the two contexts (right) obtained from `toneP2.gam`.

interval of the difference curve never contains zero, suggesting that even at the endpoint in time, the small remaining difference is still significant.

With the same method, we can also examine whether the contours of "P1.T2" and "P1.T3" are significantly different. Detailed code is provided in the supplementary material. The difference curve (left panel, Figure 13) is a straight line above zero, since "P1.T3" has higher predicted pitch values than "P1.T2" for all timepoints (right). Notably the confidence interval of the difference curve always contains zero, a clear indicator that there is no real difference between these two contexts.

In addition to the effect of context, we were interested in variation in different subject groups. Thus we further asked whether the effect of context interacts with `sex` or `location`. The next model now has two additional interactions: one between `sex` and `context`, and the other between `location` and `context`.

```
tone.gam3 = bam(pitch ~ context * (sex + location) + s(timeScaled, by=context) +
                        s(timeScaled, subject, bs="fs", m=1) +
                        s(word, bs="re"),
                data=tone, AR.start=tone$AR.start, rho=0.8, method="ML")
```

Using `compareML`, we observed a substantial improvement of model fit. Including by-sex or by-location smooths for each context, however, did not significantly improve model fit. This clarifies that in terms of tonal shape, we do not have sufficient statistical evidence supporting cross-dialect or cross-gender variation.

So far `tone.gam3` is our best model. But now we have to subject `tone.gam3` to model criticism, to clarify whether this model is actually appropriate for our data. We first checked whether residuals of this model follow a normal distribution using a quantile-quantile plot.

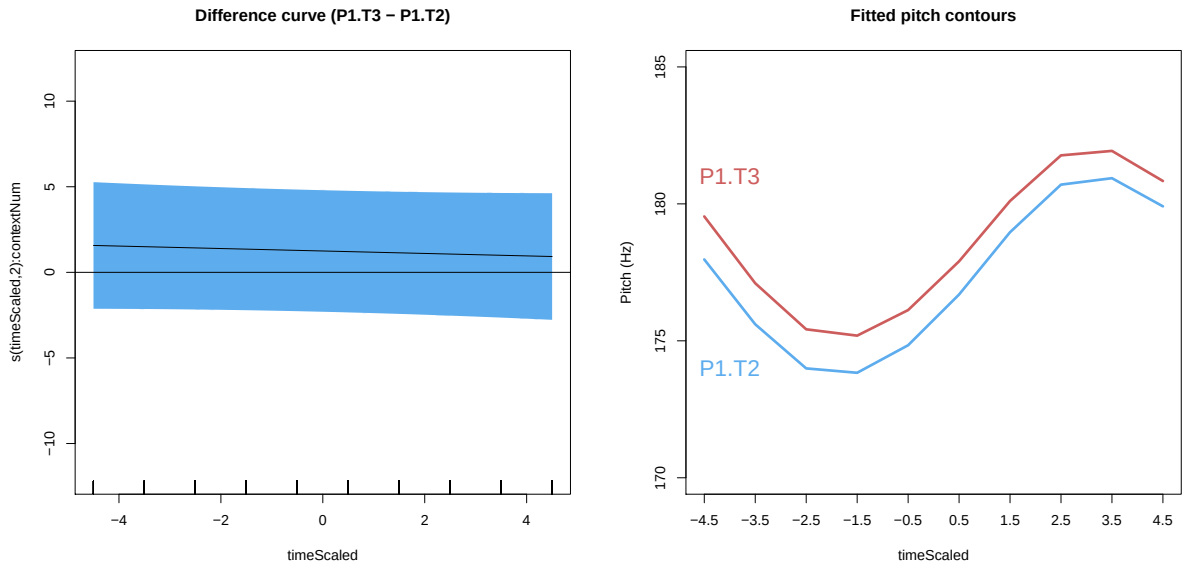

Figure 13: Difference curve between "P1.T2" and "P1.T3" contexts (left) and the predicted pitch values for the two contexts (right) obtained from model `toneP1.gam`.

```
qq.gam(gam.tone3)
```

This plot is presented in the left panel of Figure 14. Ideally, the residuals should fall on the red line, an indication that they follow a normal distribution. However, we see that the distribution of the residuals clearly has two heavy tails, indicative of a t distribution. We therefore modeled the data with a scaled-t distribution, which transforms the residuals back to normality. In the model specification we thus added `family="scat"`, which gave us `tone.gam3a`:

```
tone.gam3a = bam(pitch ~ context * (sex + location) + s(timeScaled, by=context) +
                         s(timeScaled, subject, bs="fs", m=1) +
                         s(word, bs="re"),
                 data=tone, AR.start=tone$AR.start, rho=0.8,
                 method="ML", family="scat")
```

The quantile-quantile plot for the residuals of `tone.gam3a` is presented in the right panel of Figure 14. It is clear that now the distribution of errors is much closer to a normal distribution.

A further way to carry out model criticism is to run the function `gam.check`, which provides further useful information about residuals. For example, Figure 15 presents one of the plots output by `gam.check`. It is clear that the residuals cluster into two groups. Given that `sex` is the most powerful linear predictor, the two clusters could roughly be considered as belonging to males and females. This scatterplot tells us that residuals spread out more for females (cluster on the right) than for males (cluster on the left). In other words, the model is still problematic, as it violates the assumption of homogeneity of variance for residuals. Indeed, as also shown in Figure 8, Taipei males in particular, show much less variability than Taichung males and females in general.

GAMs can directly model the variance, in addition to the mean. This can be achieved by using the Gaussian location-scale model, with the specification of `family="gaulss"`. Since we are now modeling not only the mean but also the variance, we need to specify two formulae, one for each. Combined into a list, the first formula is identical to the one of model `tone.gam3`. The second

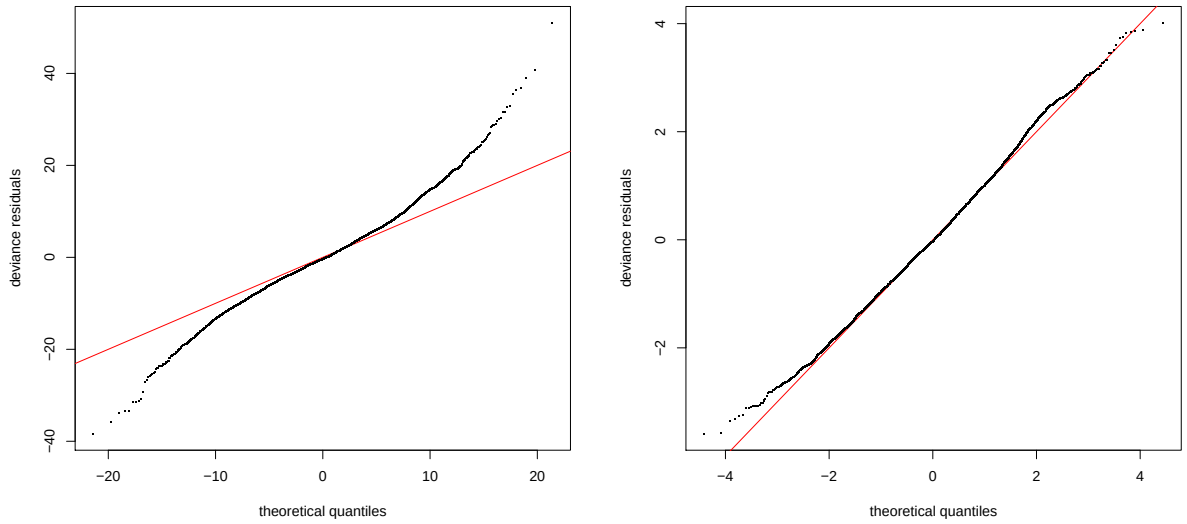

Figure 14: Quantile-quantile plots for the residuals of model `tone.gam3` and `tone.gam3a`.

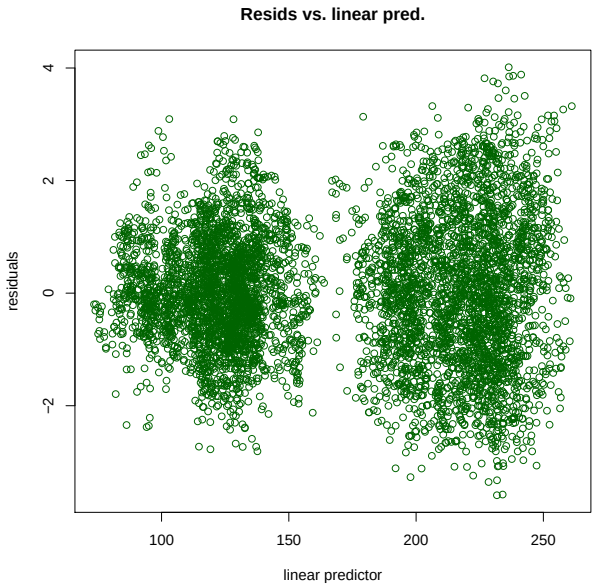

Figure 15: One of the output plots given by `gam.check` for `tone.gam3a`.

formula is for the variance, which includes an interaction between `sex` and `location`, and a main effect of `position`. We also removed the random effect of by-subject factor smooth, because subject variability coincides with the effect of `sex` × `location` to a substantial extent.

```
tone.gam4 = gam(list(pitch ~ context * (sex + location) +
                             s(timeScaled, by=context) + s(word, bs="re"),
                           ~ sex * location),
                data=tone, family="gaulss")
```

```
Parametric coefficients:
                             Estimate Std. Error z value Pr(>|z|)
(Intercept)                123.68768    0.78478 157.608  < 2e-16
contextP2.T1                 1.80746    1.10982   1.629 0.103396
...                              ...        ...     ...      ...
(Intercept).1                2.27445    0.01878 121.100  < 2e-16
sexF.1                       0.59782    0.02565  23.308  < 2e-16
locationTAICHUNG.1           0.81179    0.02657  30.557  < 2e-16
sexF:locationTAICHUNG.1     -0.62281    0.03687 -16.891  < 2e-16
```

The summary of the parametric predictors now contains two parts. The first part presents the results for modeling the mean, which is similar to the results of model `tone.gam3`. The second part is the results for modeling the variance. All three predictors reach significance, confirming our observation that the variance indeed differs across speaker groups. The downside of using the Gaussian location-scale model, however, is that autocorrelation is no longer under control. Here we have to wait for further development of GAMM theory and implementation.

In summary, the analyses of pitch contours in this section illustrate how GAMs can be used to model time series of acoustic measurements. In partilcular, we addressed the issue of autocorrelation. We also introduced difference curves, which are useful to track the difference between the curves of two contrast levels. Finally, we showed how model criticism can lead to the selection of a different "family" from the general linear model.

5 Geographic phonetic variation

To investigate the effect of dialect, traditionally it is common to treat different geographical regions as a categorical factor, and compare how and whether participants from different locations behave differently. However, location can be quantified more precisely by geographical coordinates, namely, longitude and latitude. As will be shown below, dialectal variation can be studied by fitting a nonlinear surface projected from longitude and latitude.

As mentioned previously in Section 3, retroflex sibilants are merging with dental sibilants in Taiwan Mandarin. However, the degree of merging is heterogeneous across Taiwan. In some areas, the retroflex-dental distinction still persists, whereas in other areas, sibilants are merged to a greater or lesser extent. In order to investigate how sibilant realizations differ geographically, a total of 323 participants from 117 different locations of Taiwan were recruited (Chuang et al., 2019). As shown in Figure 16, a large number of the participants are from the north, comprising the urban and suburban areas of Taipei, the capital city. From central to southern Taiwan, we only had participants from the western side of the country (between the two big cities of Taichung and Kaohsiung). This is because the eastern part is mountaneous and sparsely populated. For each geographical location, we obtained its longitude and latitude coordinates.

Stimuli were fifteen retroflex-initial and fifteen dental-initial words. The production of these words was elicited by means of a picture-naming task. For each sibilant, we measured its centroid frequency, which was calculated based on a 30-ms spectral slice extracted from the middle of the frication part of the sibilant. Tokens accompanied by non-verbal articulation such as coughing and laughing were excluded. Centroid frequency is generally considered a good index for sibilants' place of articulation: the more anterior the articulation, the higher the centroid frequency. Thus, dental articulation results in higher mean centroid frequency values compared to retroflex articulation. The pertinent information can be found in the dataset `mer`.

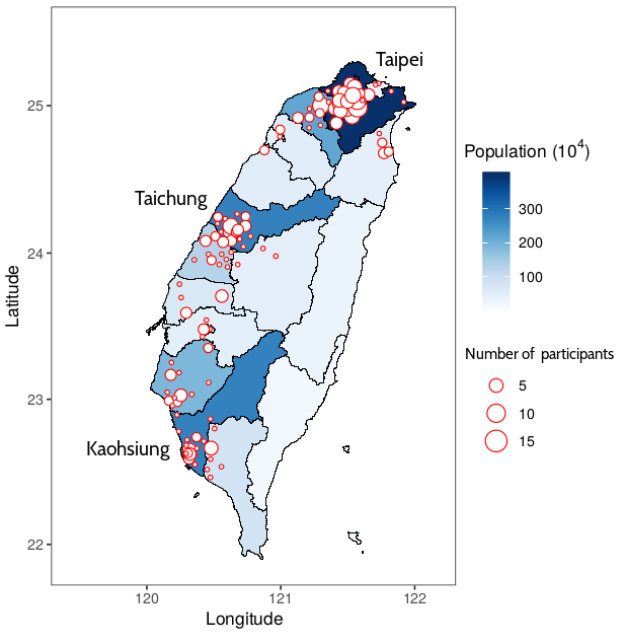

Figure 16: The geographical distribution of population in Taiwan, and the participants in this study.

```
head(mer, 3)

  Subject Word gender     vowel sibilant centfreq longitude latitude minflu
1      S1   W1      m unrounded        D 6310.199  121.5717 25.03062      6
2      S2   W1      m unrounded        D 5245.757  121.4871 25.06282      1
3      S3   W1      m unrounded        D 7276.224  121.5317 24.93039      5
```

We first fitted a model with three factorial predictors as fixed effects. In addition to `sibilant`, we also included `gender` and `vowel`, the latter of which specifies whether the sibilant is followed by a rounded or an unrounded vowel. Males' sibilants in general have lower centroid frequencies because vocal tract length is usually longer for males than for females. Similarly, when sibilants are followed by rounded vowels, the vocal tract is lengthened due to anticipatory coarticulation of lip protrusion. This leads to the lowering of centroid frequencies as well.

To inspect the effect of geography, since a geographic location is determined simultaneously by two predictors, longitude and latitude, we used **tensor product smooths**, which fit a wiggly (hyper)surface for the effects of two or more predictors. For the present dataset, a wiggly surface is projected from longitude and latitude. In addition, since we do not expect cross-region variation for dental sibilants, we added `by=sibilant` in the specification of the tensor product smooth, to request two wiggly surfaces, one for each sibilant category. This model includes by-subject and by-word random intercepts as well.

```
mer.gam0 = bam(centfreq ~ gender + vowel + sibilant +
                          te(longitude, latitude, by=sibilant) +
                          s(Subject, bs="re") + s(Word, bs="re"),
               data=mer, discrete=TRUE)
```

```
Parametric coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)      6252.8      143.9  43.446  < 2e-16
genderf           317.9       59.1   5.379 7.67e-08
vowelunrounded    759.3      112.2   6.769 1.37e-11
sibilantR       -1185.5      583.9  -2.030   0.0424

Approximate significance of smooth terms:
                                     edf  Ref.df      F  p-value
te(longitude,latitude):sibilantD   3.001   3.001  1.446    0.227
te(longitude,latitude):sibilantR  15.058  16.259  5.593 1.47e-12
s(Subject)                       295.054 323.000 23.399    0.317
s(Word)                           28.322  29.000 56.137  < 2e-16

R-sq.(adj) =  0.543   Deviance explained = 55.8%
fREML =  86336  Scale est. = 6.2616e+05  n = 10613
```

The table of parametric coefficients shows that centroid frequencies are higher for females and for sibilants followed by unrounded vowels, as expected. The effect of `sibilant`, on the other hand, is not very well supported, presumably because of merging. With regards to the effect of geography, there is also evidence supporting that retroflex (but not dental) realizations differ across regions. Before visualizing this effect, we first check the **concurvity** of the model. Concurvity is a concept similar to collinearity in linear regression. It occurs when predictors are highly correlated with one another. When there is high correlation among the predictors, coefficient estimates become inaccurate, and it becomes unclear what the unique contribution of a predictor to the model fit is. The concurvity of the model can be obtained with the function `concurvity`:

```
concurvity(mer.gam0)
```

```
         para te(longitude,latitude):sibilantD te(longitude,latitude):sibilantR
worst       1                                1                                1
observed    1                                1                                1
estimate    1                                1                                1
         s(Subject) s(Word)
worst        1.0000  1.0000
observed     0.0700  0.0114
estimate     0.0824  0.1065
```

Concurvity values are bounded between zero and one, with larger values indicating high concurvity. The fact that the estimates of the two tensor product smooths are both one suggests a serious concurvity problem of this model. The problem results from the high correlation between the effect of geographical location and the by-subject random effect. Given that for half of the locations we have only one participant, the effect of geographical location is inevitably closely tied to the effect of subject. To remedy this, we left out the by-subject random intercept, an effect which also turned out to be not statistically supported, in model `mer.gam0a`.

To visualize the regression surfaces, we use the `vis.gam` function. This function produces a contour plot of the fitted values. The output contour plots for dentals and retroflexes, overlaid with the map of Taiwan, are presented in Figure 17. More coding details are provided in the supplementary material. Warmer colors represent higher centroid frequencies, whereas colder colors represent lower centroid frequencies. We can immediately see that, as expected, dentals (left) have

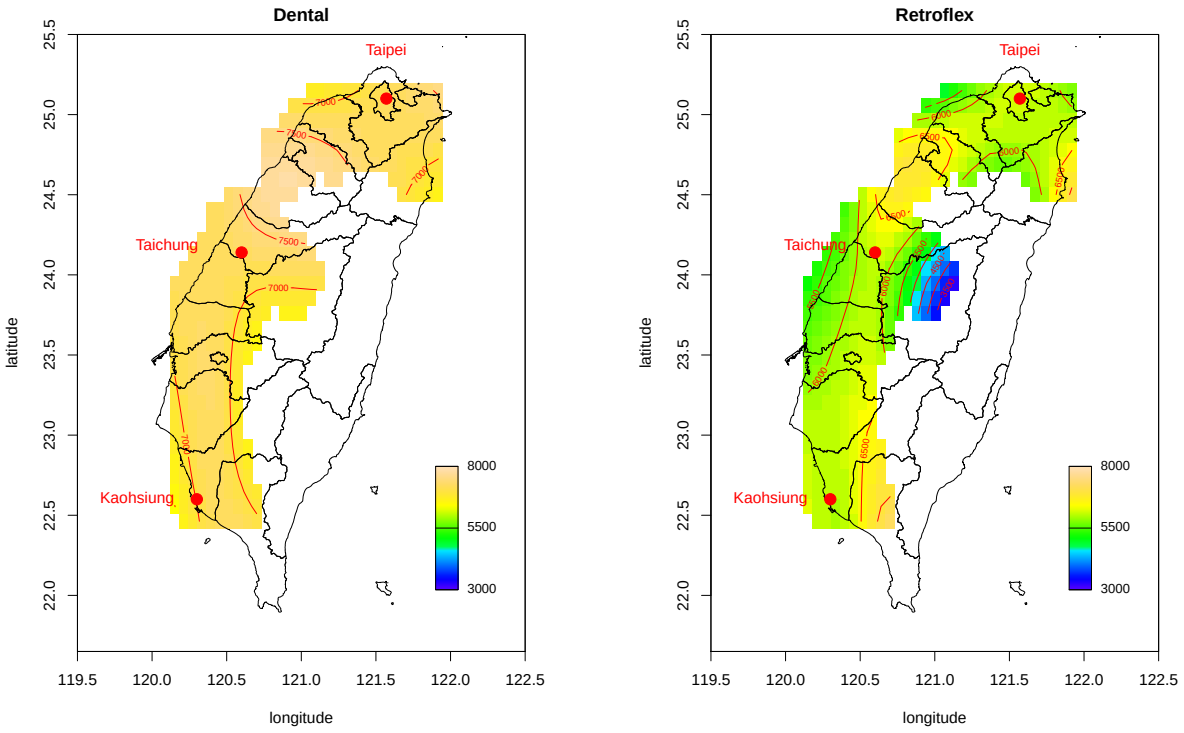

Figure 17: Geographical effect on dental (left) and retroflex (right) sibilants of `mer.gam0a`.

higher centroid frequencies than retroflexes (right). In addition, geographical differentiation is more prominent for retroflexes than dentals as it is the realization for retroflex sounds that varies substantially across regions. Specifically, people in central Taiwan (the blue area near Taichung) have very retroflexed productions, and there are a few places (the yellowish areas) where people have almost dental-like, or deretroflexed productions, indicating a high degree of merging.

Figure 17 shows that centroid frequencies vary geographically in ways that differ between dental and retroflex sibilants. This leads to the question of where across the country the dental and retroflex realizations are most similar (i.e., high degree of merging). We can investigate this by setting up a model with a difference surface, which will be predicted by the geographical coordinates (longitude and latitude). Similar to the difference curve described earlier for pitch contours in Section 4, we first transformed **`sibilant`** into a numeric variable. Next in the formula we specified two tensor products. The first one is for the reference sibilant category (dental), while the second one is for the difference surface (retroflex − dental).

```
mer$sibNum = as.numeric(ifelse(mer$sibilant=="D",0,1))
mer.gam1 = bam(centfreq ~ gender + vowel +
                          te(longitude, latitude) +
                          te(longitude, latitude, by=sibNum) +
                          s(Word, bs="re"),
               data=mer, method="ML")
```

Figure 18 presents the partial effects of the two tensor product smooths. The left panel shows the contour plot of the reference level (i.e., dental). The numbers on the contour lines are the predicted partial effect for centroid frequency. Since partial effects exclude intercepts and the effects of other predictors, we therefore observe the "pure" effect of geography. For example, Taichung is located

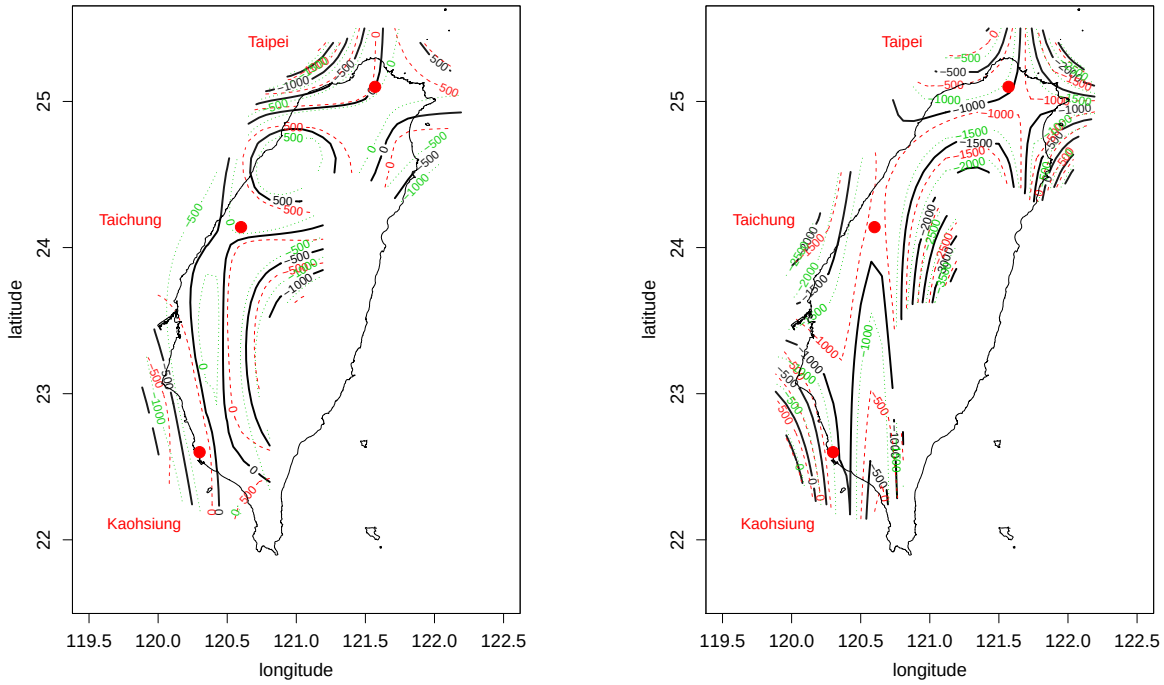

Figure 18: Contour plots of the partial effects of the reference dental sibilants (left) and the difference surface (right) obtained from model `mer.gam1`. One SE confidence regions are indicated by green dotted and red dashed lines for the upper and lower bounds of the interval respectively.

around the countour line of zero. To the east of Taichung, the number gradually decreases (from -500 to -1000). This indicates that the centroid frequency predicted for dental sibilants decreases to the east. The contour plot in the right panel presents the magnitude of the difference in centroid frequencies between the two sibilants (retroflexes − dentals). Given that retroflexes have lower centroid frequencies than dentals, more negative values thus indicate a larger difference and less merging. Notably, speakers from central Taiwan (to the east of Taichung) keep the production of the two sibilants most distinct (the predicted difference in centroid frequencies can be as large as 3000 Hz). As this is the location of the former capital of Taiwan, speakers from that region appear to preserve retroflexion to a greater extent. In addition, we found that the degree of merging for the three major cities in Taiwan is fairly similar (all located around the -1000 contour lines). This could be a consequence of urbanization. Since the population of these cities is all composed of speakers from different regions across the country, extensive interactions among different dialectal or ethnic groups could have led to a new variety that is specific to urban areas. However, we will need more data to verify this hypothesis.

One frequently asked question regarding sibilant merging in Taiwan Mandarin is to what extent the degree of merging is subject to the influence of Min, another major substrate language spoken in Taiwan. Like Mandarin, Min also has the three dental sibilants. However, unlike Mandarin, it lacks all of the three retroflex counterparts. Merging, essentially deretroflexion, is thus often regarded as a negative transfer from Min. The question of interest is whether Min fluency has an effect on sibilant merging. Specifically, we ask whether highly fluent Min speakers deretroflex more when speaking Mandarin, and also whether knowledge of Min interacts with the geographic effect.

In the `mer` dataset, the column `minflu` provides our participants' self-reported ratings of Min

speaking fluency. The ratings range from one (low fluency) to seven (high fluency). In the next model, we included a tensor product for `minflu` and the two geographical coordinates, and further requested separate wiggly surfaces for the two sibilant categories. We fitted the model with the maximum likelihood method to enable model comparisons.

```
mer.gam2 = bam(centfreq ~ gender + vowel + sibilant +
                          te(longitude, latitude, minflu, by=sibilant) +
                          s(Word, bs="re"),
               data=mer, method="ML")
```

Compared to `mer.gam0a` (refitted with `method="ML"`), `mer.gam2` clearly has better model fit. To visualize the interaction of longitude and latitude by Min fluency, we plot separate maps for different levels of Min fluency. Given that most of our participants have Min proficiency ratings between three and six, we therefore focused on these four fluency levels.

Figure 19 presents the effect of the four-way interaction, with the predicted surface for dentals in the upper row, and the predicted surface for retroflexes in the bottom row. Columns are tied to Min fluency, increasing from three (left) to six (right). At all Min fluency levels, the distinction between the two sibilants is still retained. Dental realizations are comparatively stable, whereas retroflex realizations vary across Min fluency levels to a much greater extent. Generally speaking, speakers from the center of Taiwan have more retroflexed pronunciation, but typically for speakers with mid-to-high Min fluency. On the other hand, for speakers from northern Taiwan (areas with latitude above 24.5), the realization of retroflexes becomes more dental-like with increasing Min fluency, consistent with the hypothesis of negative Min transfer. Interestingly, our analysis clarifies that this influence is region-specific.

In summary, we have shown how to analyze the geographical distibution of sociophonetic variation with GAMs. Instead of using categorical variables that distinguish different geographical locations, we showed that directly working with geographical coordinates provides us with novel and powerful ways of studying variation. Specifically, tensor product smooths are available for modeling nonlinear interactions between two or more predictors, enabling dialectal variation to be studied in conjunction with other phonetic or social variables in considerable detail.

6 Further references

This chapter provides only a basic introduction to the main concepts of GAMs. We refer the interested reader to Wood (2017) for a detailed exposition of the mathematics underlying GAMs and discussion of a wide range of examples of application to empirical data. A general, non-technical introduction to the mathematical concepts underlying GAMs is provided by Baayen et al. (2017).

Introductions to GAMs focusing specifically on application to phonetics are Wieling et al. (2016); Baayen and Linke (2019). Wieling et al. (2016) used GAMs to analyze tongue movement data obtained with electromagnetic articulography, investigating the difference in articulatory trajectories between two Dutch dialects. Baayen and Linke (2019) analyzed the distribution and occurrence of pronunciation variants, extracted from the Buckeye corpus, with GAMs. A GAM analysis on eye-tracking data collected for the perception of Cantonese tones can be found in Nixon et al. (2016).

Quantile GAMs (QGAMs Fasiolo et al., 2020) provide an extension of the GAM framework that makes it possible to study how the quantiles of the distribution of a response variable depend on a set of predictors. As QGAMs are distribution-free, they are especially useful for datasets for which models that build on the assumption that the residual errors should be independently and identically distributed turn out to be inappropriate. For this reason, Tomaschek et al. (2018) turned to QGAMs for modeling tongue movements registered with electromagnetic articulography.

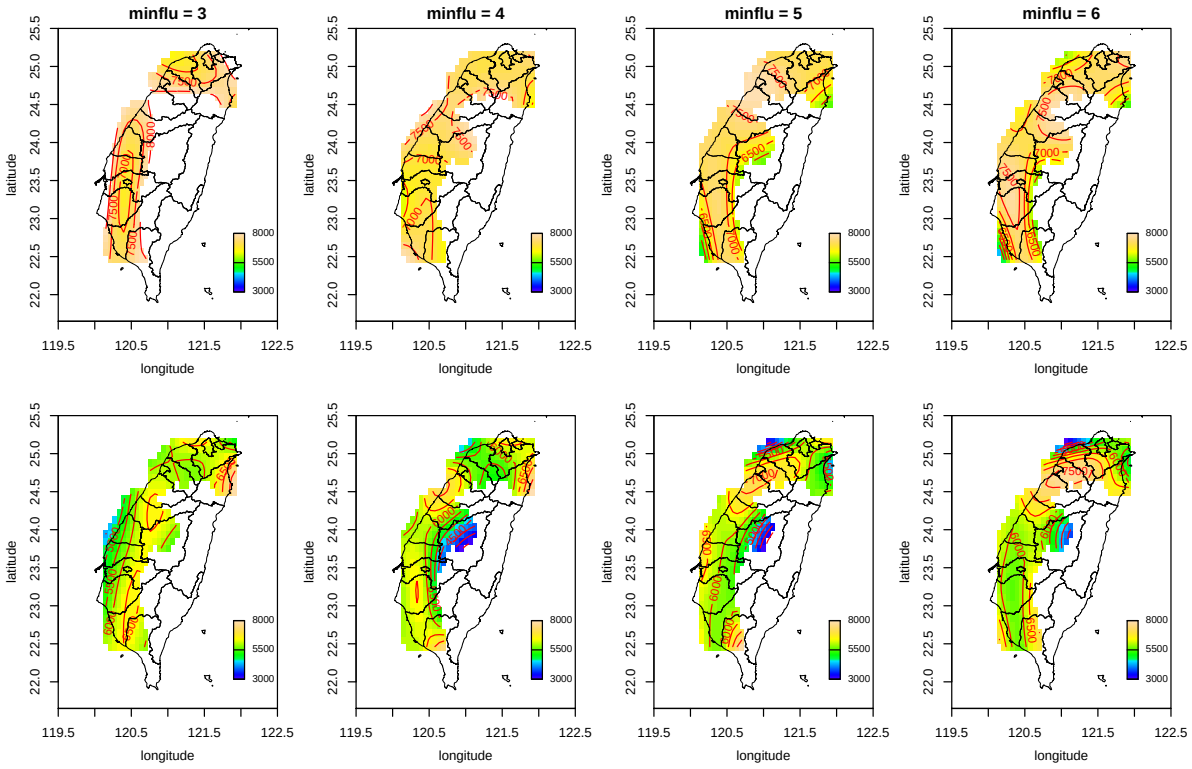

Figure 19: The effect of geography on dentals (upper panels) and retroflexes (lower panels), conditioned on Min fluency levels.

References

Baayen, R. H. and Linke, M. (2019). An introduction to the generalized additive model. In Gries, S. T. and Paquot, M., editors, *A practical handbook of corpus linguistics*. Springer, Berlin. (forthcoming).

Baayen, R. H., Vasishth, S., Kliegl, R., and Bates, D. (2017). The cave of shadows: Addressing the human factor with generalized additive mixed models. *Journal of Memory and Language*, 94:206–234.

Chuang, Y.-Y. (2017). *The effect of phonetic variation on word recognition in Taiwan Mandarin*. PhD thesis, National Taiwan University, Taipei.

Chuang, Y.-Y., Sun, C.-C., Fon, J., and Baayen, R. H. (2019). Geographical variation of the merging between dental and retroflex sibilants in Taiwan Mandarin. In *Proceedings of the 19th International Congress of Phonetic Sciences*, pages 472–476, Melbourne. Australasian Speech Science and Technology Association.

Fasiolo, M., Wood, S. N., Zaffran, M., Nedellec, R., and Goude, Y. (2020). Fast calibrated additive quantile regression. *Journal of the American Statistical Association*, pages 1–28.

Fon, J. (2007). The effect of region and genre on pitch range of Tone 1 in Taiwan Mandarin. Technical Report NSC95-2411-H-002-046-, National Science Council, Taipei, Taiwan.

Nixon, J. S., van Rij, J., Mok, P., Baayen, R. H., and Chen, Y. (2016). The temporal dynamics of perceptual uncertainty: eye movement evidence from cantonese segment and tone perception. *Journal of Memory and Language*, 90:103–125.

Tomaschek, F., Tucker, B. V., Fasiolo, M., and Baayen, R. H. (2018). Practice makes perfect: The consequences of lexical proficiency for articulation. *Linguistics Vanguard*, 4(s2).

Tucker, B. V., Brenner, D., Danielson, D. K., Kelley, M. C., Nenadić, F., and Sims, M. (2018). The Massive Auditory Lexical Decision (MALD) database. *Behavior research methods*, pages 1–18.

van Rij, J., Wieling, M., Baayen, R. H., and van Rijn, H. (2017). itsadug: Interpreting time series and autocorrelated data using GAMMs. R package version 2.3.

Wieling, M. (2018). Analyzing dynamic phonetic data using generalized additive mixed modeling: a tutorial focusing on articulatory differences between L1 and L2 speakers of English. *Journal of Phonetics*, 70:86–116.

Wieling, M., Tomaschek, F., Arnold, D., Tiede, M., Bröker, F., Thiele, S., Wood, S. N., and Baayen, R. H. (2016). Investigating dialectal differences using articulography. *Journal of Phonetics*, 59:122–143.

Wood, S. (2017). *Generalized Additive Models: An Introduction with R*. Chapman and Hall/CRC, 2 edition.