A tutorial on Generalised Additive Mixed Effects Models for bilingualism research

Stefano Coretta, Joseph V. Casillas

Motivation

In recent years quantitative analysis of linguistic data has benefited greatly from the now commonplace use of so-called random effects (also known as hierarchical, mixed-effects, multi-level or nested models, Winter (2013); Winter (2020)). While this shift represents a clear improvement in statistical analyses, particularly in areas of linguistics in which nested structure is the norm (e.g., trial repetitions nested within participants), an over-reliance on standard linear modelling prevails, particularly in the cases of dynamic phenomena that may not constitute a linear relationship, e.g., vowel trajectories, pitch contours, acquisition processes, etc.

Generalised Additive (Mixed) Models (GAMMs) are now commonly employed in phonetic research (given the naturally dynamic nature of speech data) and this is reflected by the availability of several tutorials which focus on phonetic data (Sóskuthy, 2017, 2021; Tamminga et al., 2016; Wieling, 2018). This is why we hope that this tutorial will make GAMMs accessible to researchers from other fields within linguistics. In particular, this tutorial is thought for researchers in bilingualism and multilingualism who wish to be able to start using GAMMs for non-linear data, which is very common in developmental and learning phenomena.

Generalised additive (mixed) models

Generalised Additive Models (GAMs) and their mixed-effects version (GAMMs) represent a variation of the generalised linear model that can accommodate modelling non-linear relationships (Hastie & Tibshirani, 1986; Wood, 2006). The main component that is specific to GAMMs are smoothers or smooth terms. These are model terms that can fit non-linear effects. An advantage of using smooth terms is that they are constrained by a penalty measure—part of the fitting procedure and does not depend on user input—which prevents over-fitting the data. This approach is different from, for example, growth curve analyses which are based on user-specified polynomial degrees.

Readers interested in a more detailed conceptual discussion of GAMMs are referred to Sóskuthy (2017). The rest of this tutorial focuses on applying GAMMs in bilingualism research using the statistical programming language R (R Core Team, 2023).

Pre-requisites

In what follows, we detail the packages needed to complete the tutorial, followed by two specific case studies designed to illustrate how GAMMs can be applied to model common phenomena in second language acquisition and bilingualism research.

This tutorial assumes readers are already familiar with R (R Core Team, 2023) and have at least some experience fitting linear models, including models with random effects. The following packages need to be installed:

- tidyverse

- mgcv

- tidygam

You can find the tutorial code and data here: https://github.com/stefanocoretta/gamm_biling.

Case study 1: U-shaped learning

U-shaped learning

Research on language acquisition first documented over-regularization errors in the 60's (Cazden, 1968; Ervin & Miller, 1963). Subsequent research found that over-regularization errors tended to manifest in a developmental curve, and is now referred to as the *U-shaped developmental curve* (Williams et al., 2022). In most cases, U-shaped development is considered to be a three-step process, which begins with accurate performance, followed by a period in which performance dips, and then subsequently becomes accurate again (Carlucci & Case, 2013). Interestingly, U-shaped curves are observed in numerous cognitive-developmental and learning contexts, particularly with regard to child development, e.g., understanding temperature, weight conservation, object permanence, and facial recognition (See Carlucci & Case, 2013), which, in turn, has fostered fruitful research and debate in Cognitive Science regarding learning models (e.g., Marcus et al., 1992; Rumelhart & McClelland, 1986, among others).

In first language acquisition research, the most cited example stems from children's use of the past tense (Cazden, 1968; Ervin & Miller, 1963; Marcus et al., 1992). For instance, Cazden (1968) found that children accurately produced irregular past tense verbs during a period of time, and, at a later point in time, began to produce over-generalization errors, e.g., *feet* followed by **feets*. Eventually past tense production again surfaces as expected, typically around

the age of three, which coincides with the developmental stage in which children acquire regular past tense verbs (See Marcus et al., 1992). In the realm of Second Language Acquisition (SLA), researchers have documented non-linear, U-shaped courses of development in numerous areas for both developmental and instructional sequences (See Casani, 2020; Geeslin & Guijarro-Fuentes, 2006; Kellerman, 1985; Long, 1990; Pliatsikas & Marinis, 2013; Shirai, 1990; Williams et al., 2022, among many others). Kellerman (1985), for instance, documented Dutch learners of English acquiring verb alternation patterns of English following a U-shaped trajectory. In a similar vein, Lightbown (1983) showed accuracy producing progressive forms (i.e., *-ing*) followed a U-shaped trajectory when acquiring the simple present/present continuous tense/aspect distinction in French speaking learners of English. U-shaped learning may be a by-product of reorganization or restructuring of prior knowledge (Gass & Selinker, 2008; Shirai, 1990), and likely accounts for language instructors' contention that students often regress when learning new linguistic forms. Be that as it may, it is clear that the non-linear quality of this aspect of language learning requires a non-linear approach to modelling the underlying process.

The data

We have simulated data for accuracy scores from 200 language learners, taken at 10 different time points. A proficiency score was also included for each participant at each time point. The data is intended to simulate a study in which participants perform the same learning task and proficiency assessment in a longitudinal design. We print the first six rows of the data frame below.

```
dat1 <- readRDS("data/dat1.rds")

# Print first 6 rows of the dataframe
head(dat1)
```

```
# A tibble: 6 x 4
  score proficiency subj   time
  <dbl>       <dbl> <fct> <int>
1  26.7       -1.37 s1        0
2  26.4       -1.29 s1        1
3  26.0       -1.21 s1        2
4  25.7       -1.14 s1        3
5  25.5       -1.06 s1        4
6  25.2      -0.983 s1        5
```

The `score` column contains the accuracy scores, while the `proficiency` column the proficiency scores. The participant ID is given in `subj`. time point 0 to 9 is in `time`. Figure 1 shows the

relationship between proficiency and learning scores for individual subjects. From the figure, it is clear that such relationship has a non-linear, U-shape, in which learning scores initially decrease, plateau, then increase again as proficiency increases.
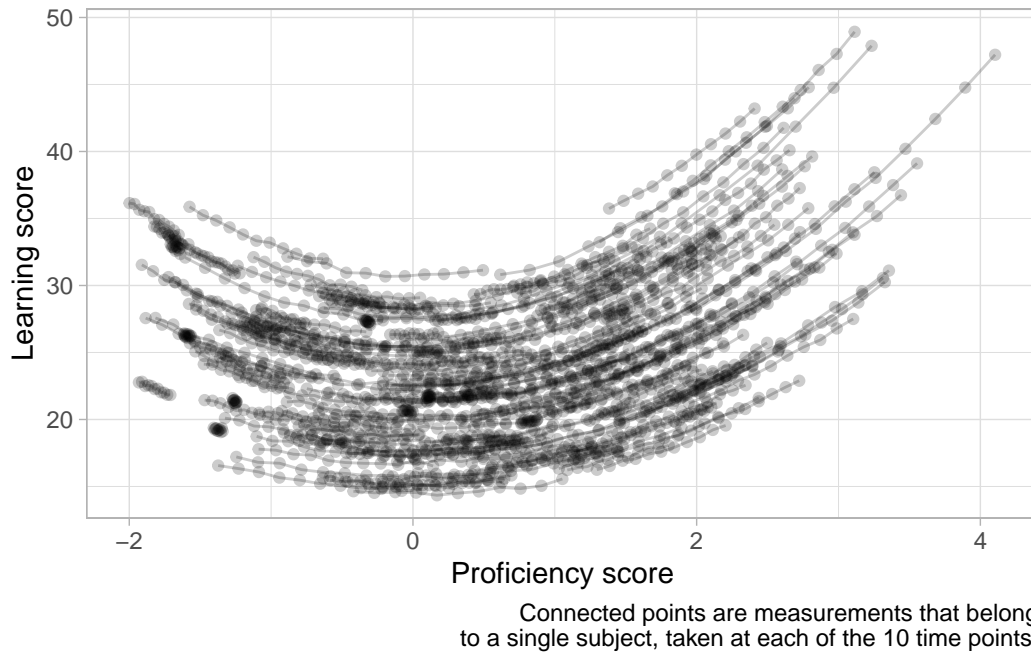

Figure 1: Scatter plot of proficiency and learning scores illustrating a U-shaped learning curve.

In the following sections we will analyse this data using GAMMs. For pedagogical purposes, we will first focus on the effect of proficiency scores on learning scores at a single time point, time point 5. Subsequently, we will analyse the entire data set, to illustrate how to conduct a time-series analysis.

Modelling a non-linear effect

Let's first focus on how to model a non-linear effect: here, we can look at the effect of the proficiency score on learner accuracy. As we have seen in Figure 1, the effect is U-shaped, i.e. it is not linear. To simplify, we will begin our analysis using data from a single time point. Later we will expand the analysis to also include time as a predictor. We model a non-linear effect of proficiency on accuracy with the following code.

```
# Subset the data to time point 5
dat15 <- filter(dat1, time == 5)

# attach the mgcv package
```

```
library(mgcv)

# fit the model
gam_1 <- gam(
  score ~ s(proficiency),
  data = dat15
)
```

The formula states that `score`, the outcome variable (also known as the dependent variable or criterion) is modelled as a function of `proficiency`. Note that we use `s()` to indicate that we want to estimate a (potentially) non-linear effect. The name of the function, `s`, stands for "smooth term". Smooth terms (aka smoothers) are mathematical functions that allow GAMs to fit non-linear effects. A detailed treatment of smoothers is beyond the scope of this tutorial. We refer interested readers to Wood (2006) for a description of the underlying math.

Before looking at the summary of the `gam_1` model, we plot the predicted effect of proficiency. For this, we will use the tidygam package, which provides users with utility functions that make extracting and plotting predictions from GAMs more accessible.

```
# attach tidygam
library(tidygam)

# extract model predictions
gam_1_preds <- predict_gam(gam_1)

# plot predictions
plot(gam_1_preds, series = "proficiency")
```
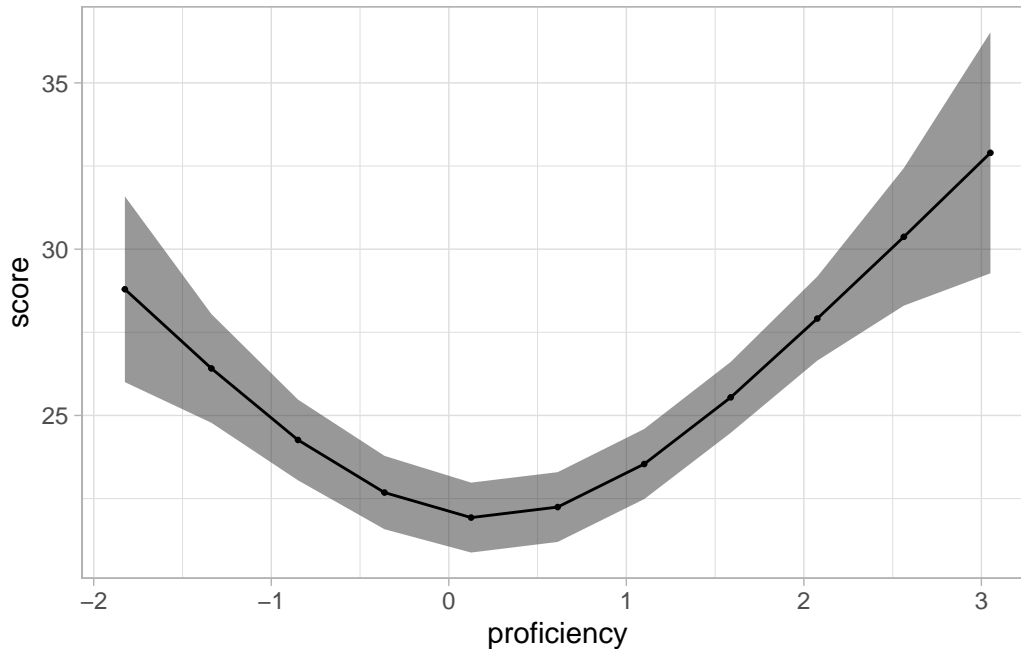
Figure 2: Predicted accuracy score depending on proficiency, from a GAM.

Figure 2 is a plot of the predicted effect of proficiency on accuracy, based on the `gam_1` model. We will look into the details of predict_gam() shortly, but for now, notice that accuracy initially decreases as proficiency increases. At proficiency of approximately 0.5, accuracy begins to increase, resulting in the typical U-shaped curve described above. At this juncture we will inspect the model summary.

```
summary(gam_1)
```

```
Family: gaussian
Link function: identity

Formula:
score ~ s(proficiency)

Parametric coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   24.559      0.331   74.19   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Approximate significance of smooth terms:
                 edf Ref.df     F p-value
s(proficiency) 2.922  3.675 16.46  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =  0.242   Deviance explained = 25.3%
GCV = 22.352  Scale est. = 21.914    n = 200
```

The relevant parts of the summary for the time being are the `Parametric coefficients:` table and the `Approximate significance of smooth terms:` table. The former contains the estimate of the intercept. This intercept represents the same intercept you would estimate in a standard linear model. In our case, the estimate of the intercept is the predicted accuracy score when proficiency is equal to zero. Thus our interpretation according to the summary, is that when proficiency is 0, learner accuracy is about 25.

Importantly, what we are interested in is the effect of proficiency on accuracy, rather than the estimate of accuracy when proficiency is zero, i.e., the intercept. Information on the effect of proficiency is found in the second table, which contains estimates of the smooth terms. Alas, these estimates are not informative about the effect, per se, but rather indicate if the relationship between proficiency and accuracy is linear (or not) in nature. More specifically, the `edf` estimate, which stands for Estimated Degrees of Freedom, is equal to one in the case of a perfectly linear relationship. A value greater than one is an indication of non-linear effects. Crucially, the EDF estimate is not informative with regard to the exact shape of the effect. The only way to assess this is to plot the model predictions, as we have done in Figure 2.

In our `gam_1` model, the `edf` of the smooth term for proficiency is 2.9 and is significantly different from 1 (p < 2e-16), thus suggesting that the effect of proficiency on accuracy is not linear (Note: the `Ref.df`, reference degrees of freedom, and `F` values merely serve the purpose of being used to derive a *p*-value).

If we were to report this model and results, we could write something along the following lines: We fitted a Generalised Additive Model to learner accuracy scores, with a smooth term over proficiency (to model non-linear effects). According to the model, the effect of proficiency is significantly non-linear ($F = 16.46$, $p < 0.0001$). Based on the prediction plot, we observe that at lower proficiency, learning scores initially decrease and then start increasing again when proficiency reaches approximately 0.5.

We began our analysis with a simplified model examining the effect of proficiency on accuracy at a single time point for pedagogical purposes. In the following section we will expand our analysis by refitting the data including time as a predictor.

Multiple smooth terms

As mentioned previously, the data `dat1` contains accuracy and proficiency scores from 200 subjects, taken at 10 time points. The data was simulated so that proficiency increased over time (at different degrees for different participants), as is often the case in language acquisition processes. An interesting question to consider is whether learning scores improve with time independently of proficiency, or if proficiency alone is causing learning scores to improve. We can approach this question using principles of causal inference. To this end, we develop directed acyclic graphs (DAGs) to describe the causal relationships involved. A full treatment of causal inference is beyond the scope of this paper. Interested readers are referred to McElreath (2019).

The DAGs in Figure 3 represent the causal relationships between accuracy, proficiency scores and time in two scenarios. In (a), time affects proficiency and proficiency affects accuracy. In other words, time has a direct effect on proficiency, but not on accuracy; accuracy scores can be predicted from proficiency alone. In (b), on the other hand, time affects proficiency, as in (a), but also accuracy (and proficiency affects accuracy scores as in (a)). The practical interpretation is that time affects accuracy in two ways: through its effect on proficiency and directly through its effect on accuracy.

(a) No direct effect of time on learning scores.   (b) Direct effect of time on learning scores.
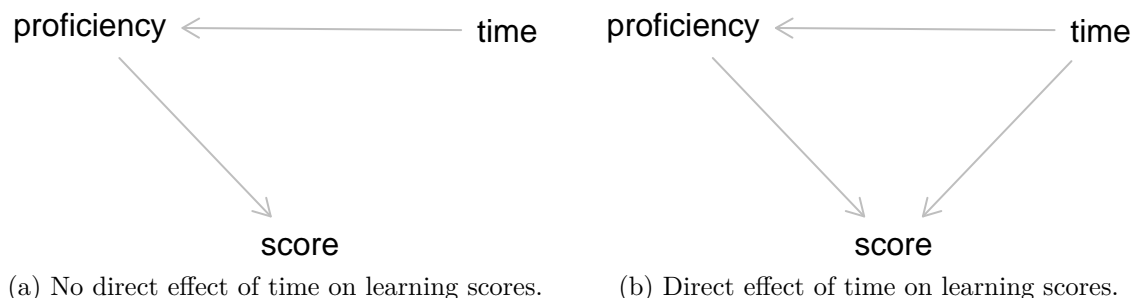
Figure 3: Dyrected Acyclic Graphs for time, proficiency and learning score.

DAGs allow us to make causal statements based on statistical results. In this case, when including both time and proficiency as predictors in a GAM, time should not have an effect on learner accuracy if scenario (a) is correct (while it should have an effect if scenario (b) is correct). Now we fit the model.

```
gam_2 <- gam(
  score ~ s(proficiency) + s(time) +
    s(subj, bs = "re"),
  data = dat1
)
```

In `gam_2` we fit a GAM to learner accuracy `score` with two predictors: a smooth term over proficiency and a smooth term over time. We also include random effects to account for

nested data from multiple participants. The syntax for random effects in GAMs is different from the more familiar syntax used in lme4. With `gam()`, we can specify random effects using smooth terms and the argument `re` (for Random Effects) basis function. Random intercepts are added with the syntax `s(ranint, bs = "re")` and random slopes with the syntax `s(ranint, ranslope, bs = "re")`. Here we only include a by-subject random intercept for illustration. Below we inspect the model summary.

```
summary(gam_2)
```

```
Family: gaussian
Link function: identity

Formula:
score ~ s(proficiency) + s(time) + s(subj, bs = "re")

Parametric coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   24.738      0.441   56.09   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
                   edf  Ref.df       F  p-value
s(proficiency)   8.699   8.976 4638.38  < 2e-16 ***
s(time)          1.000   1.000   20.27 7.36e-06 ***
s(subj)        198.943 200.000 2637.63  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =  0.997   Deviance explained = 99.7%
GCV = 0.10023  Scale est. = 0.090681  n = 2200
```

Now we plot the model predictions of accuracy score by proficiency and time point separately.

```
gam_2_preds <- predict_gam(
  gam_2, length_out = 25,
  series = "proficiency",
  exclude_terms = c("s(subj)", "s(subj,proficiency)")
)
```

```
Warning in mgcv::predict.gam(model, newdata = pred_grid, se.fit = TRUE, :
non-existent exclude terms requested - ignoring

Warning: There was 1 warning in `dplyr::mutate()`.
i In argument: `fit = rowSums(dplyr::across())`.
Caused by warning:
! Using `across()` without supplying `.cols` was deprecated in dplyr 1.1.0.
i Please supply `.cols` instead.
```
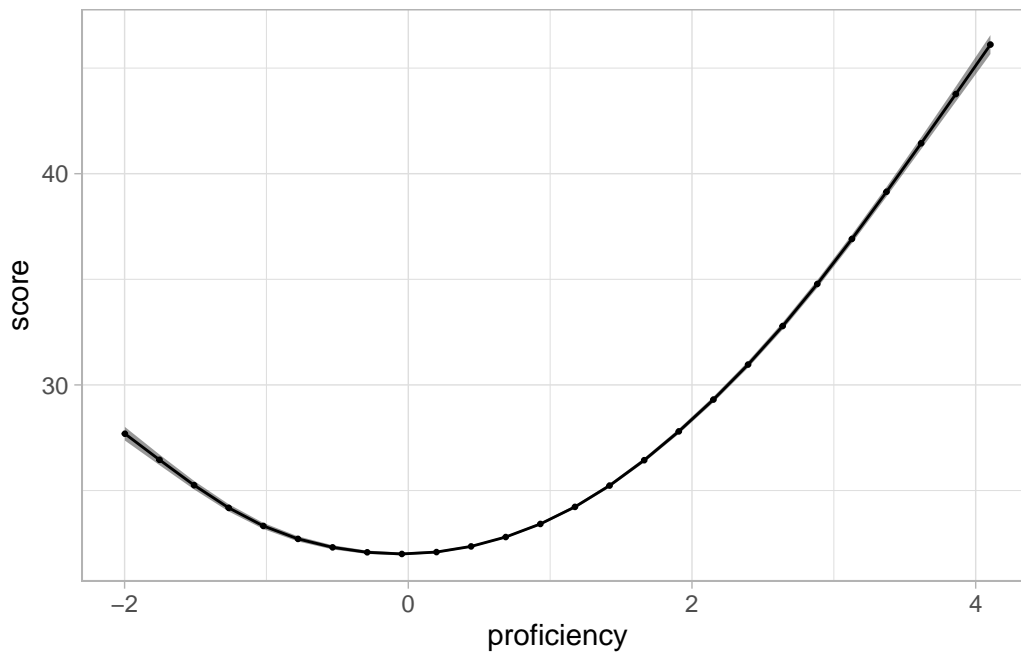
```
plot(gam_2_preds)
```


Figure 4: Predicted accuracy score as a function of proficiency, from a GAM.

```
gam_2_preds_t <- predict_gam(
  gam_2, length_out = 25,
  series = "time",
  exclude_terms = c("s(subj)", "s(subj,proficiency)")
)
```

```
Warning in mgcv::predict.gam(model, newdata = pred_grid, se.fit = TRUE, :
non-existent exclude terms requested - ignoring
```
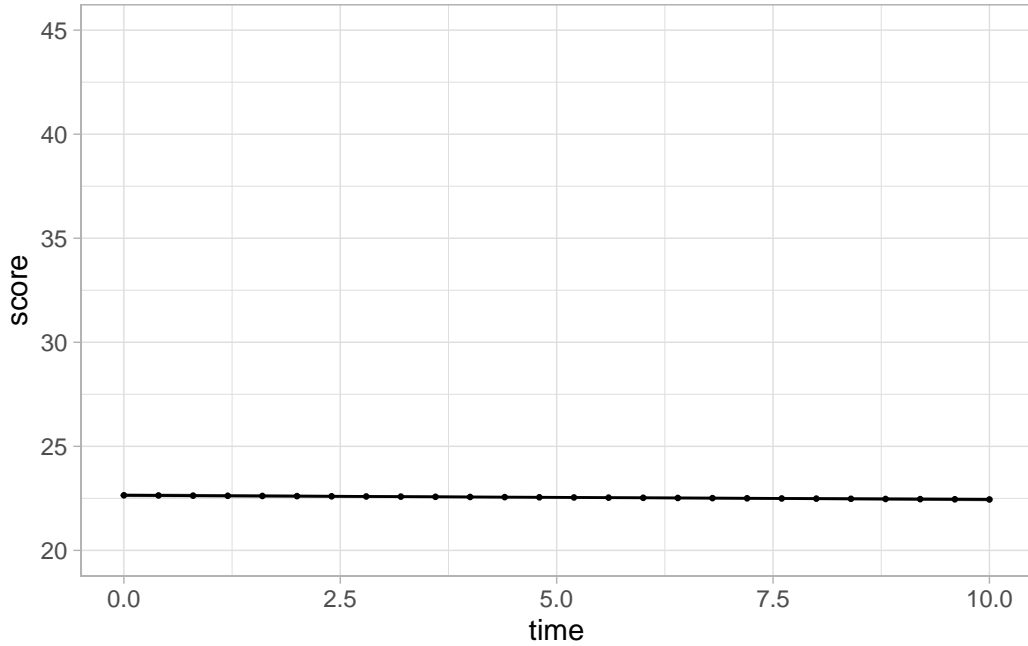
```
plot(gam_2_preds_t) +
  ylim(20, 45)
```


Figure 5: Predicted accuracy score as a function of proficiency, from a GAM.

Case study 2: L2 vowel production in simultaneous and late bilinguals

To motivate our next example, we will now consider the production of Spanish vowels in three groups of bilinguals: simultaneous/native English/Spanish bilinguals (henceforth `es`) and late learners of Spanish with low and high levels of proficiency, `es_0` and `es_1`, respectively. Spanish has five phonemic vowels, /i, e, a, o, u/, and is often described as being a syllable-timed language in which the spectral envelope is rather stable and stressed and unstressed syllables have approximately the same duration. Though there is evidence that unstressed vowels can centralize to a certain degree, the differences between stressed and unstressed vowels are believed to be imperceptible (Martínez Celdrán, 1984). American English, on the other hand, has a larger vowel inventory and is often described as a stress-timed language in which the duration between stressed syllables is approximately the same. Vowel reduction can occur in unstressed syllables, which typically manifests via shortening and/or centralization, with unstressed vowels often reducing to schwa ([ə]). Accordingly, English speaking learners of Spanish face a substantial obstacle when it comes to producing and perceiving Spanish vowels (See Aldrich, 2014; Bland, 2016; Cobb, 2009; Cobb & Simonet, 2015; Iruela, 1997, among others). To wit, they often display cross-linguistic influence by producing unstressed

Spanish vowels with [ə], particularly in the case of /a/, e.g., "casa" (Eng. *house*) as [ka.sə], and diphthongizing vowels in final position (Cobb & Simonet, 2015).

In the following example, we consider the production of Spanish vowels in the aforementioned groups of bilingual participants. Interestingly, to our knowledge, all of the current research investigating the cross-linguistic influence of vowel reduction processes in learners of Spanish have utilized modelling strategies that rely upon formant measurements at a single time point, typically the mid-point of the vowel, rather than scrutinizing the entire trajectory of the spectral envelope. In our view, the dynamic nature of vowel formants poses an interesting use-case for modelling using non-linear methods, such as GAMs. To this end, we have simulated phonetic data from three aforementioned groups of bilinguals: simultaneous English/Spanish bilinguals, beginner late English/Spanish bilinguals and advanced late English/Spanish bilinguals. The data contains the Euclidean Distance (EuD) of three Spanish vowels /a, i, u/ from the vowel space centroid, taken from word-final non-stressed vowels, along nine equidistant time points. The measure is a good proxy for vowel reduction phenomena, which is to be expected in late bilinguals.

```
dat2 <- readRDS("data/dat2.rds")

head(dat2)
```

```
# A tibble: 6 x 7
  subj  group vowel   rep   eud timep id
  <fct> <chr> <chr> <int> <dbl> <int> <chr>
1 es_1  es    i         1  1.85     1 es.1.1
2 es_1  es    i         1  1.86     2 es.1.1
3 es_1  es    i         1  1.83     3 es.1.1
4 es_1  es    i         1  1.87     4 es.1.1
5 es_1  es    i         1  1.89     5 es.1.1
6 es_1  es    i         1  1.81     6 es.1.1
```

This is what the data looks like. The top row reports the synthetic data of simultaneous English/Spanish bilinguals (`es`). It can be observed that the Euclidean distance is quite stable across the duration of the vowel for all three vowels and away from 0 (which corresponds to a mid-central vowel), indicating no reduction takes place (as it is to be expected for Spanish). On the other hand, in the second and third row, we can see some vowel reduction at play: for /a/, beginners late bilinguals (`es_0`) completely reduce the vowel to a mid-central vowel, while the advanced late bilinguals (`es_1`) reduce it to a lesser degree, but it is still not quite a full Spanish /a/; /i/ and /u/ are produced with a diphthongal quality, where the first part of the vowel is quite reduced. Finally, comparing beginners and advanced bilinguals shows that the latter have less reduction, as expected by increasing proficiency in Spanish.
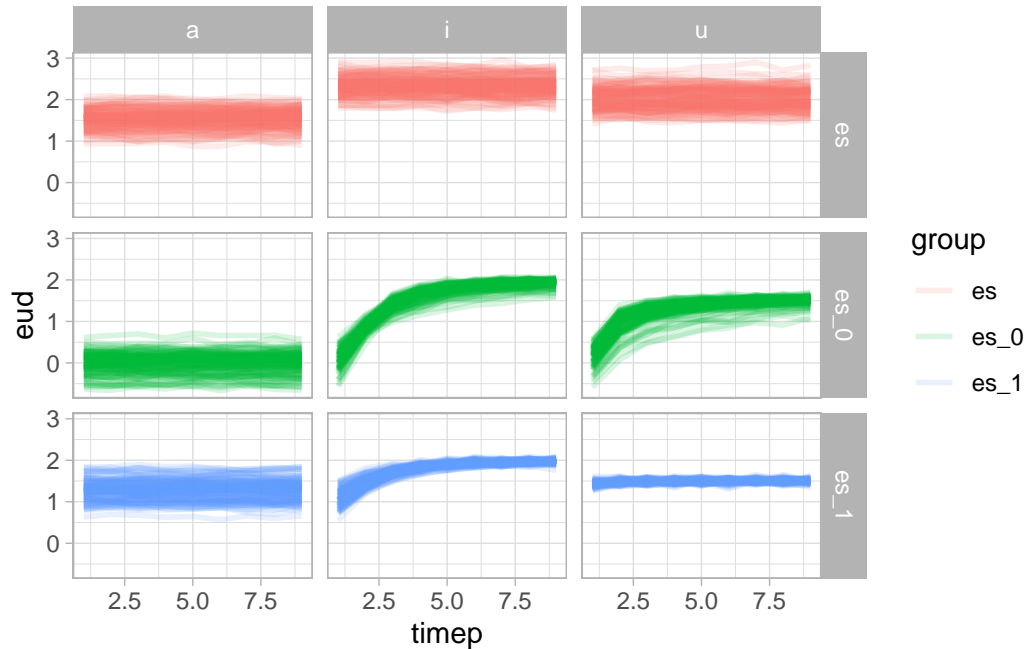
Figure 6: Euclidean Distance through 9 time points taken from within vowels.

We can use a GAM to model Euclidean distance along time point in the different vowels and different bilingual groups. Fitting smooth terms for different levels of categorical predictors can be achieved by specifying the categorical predictor as the value of the `by` argument inside the smooth term function `s()`: for example `s(timep, by = vowel)` would ensure a non-linear effect of time point is estimated for each vowel. In our data, we want to model the interaction of vowel (/a, i, u/) and group (`es, es_0, es_1`). Unfortunately, classical interactions cannot be model in GAMs (because of the "additive" aspect of the model; interactions require product operations).

To obviate this limitation, we can simply construct a new predictor with the combination of vowel and group and use that as the `by`-variable in the smooth terms. Due to how smooth terms are constructed when you include a `by`-variable, it is necessary to also include the variable as a parametric term (i.e. a classical linear term).

```
dat2 <- dat2 %>%
  mutate(
    vowel = as.factor(vowel),
    group_vowel = interaction(group, vowel),
  )
```

13

```r
gam_3 <- gam(
  eud ~
    # Parametric term for group_vowel
    group_vowel +
    # Smooth term with group_vowel by-variable
    s(timep, by = group_vowel, k = 9) +
    # Random factor smooths
    s(timep, subj, bs = "fs", m = 1),
  data = dat2
)

summary(gam_3)
```

```
Family: gaussian
Link function: identity

Formula:
eud ~ group_vowel + s(timep, by = group_vowel, k = 9) + s(timep,
    subj, bs = "fs", m = 1)

Parametric coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)         1.533926   0.023247  65.983  < 2e-16 ***
group_voweles_0.a  -1.539712   0.032877 -46.833  < 2e-16 ***
group_voweles_1.a  -0.255481   0.032877  -7.771 8.59e-15 ***
group_voweles.i     0.791406   0.007385 107.157  < 2e-16 ***
group_voweles_0.i  -0.052418   0.032877  -1.594    0.111
group_voweles_1.i   0.222520   0.032877   6.768 1.38e-11 ***
group_voweles.u     0.485860   0.007385  65.786  < 2e-16 ***
group_voweles_0.u  -0.318526   0.032877  -9.689  < 2e-16 ***
group_voweles_1.u  -0.044394   0.032877  -1.350    0.177
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
                            edf  Ref.df       F p-value
s(timep):group_voweles.a   1.000   1.000   0.112 0.73770
s(timep):group_voweles_0.a 1.000   1.000   0.117 0.73226
s(timep):group_voweles_1.a 1.000   1.000   0.023 0.87934
s(timep):group_voweles.i   1.000   1.000   0.021 0.88372
```

```
s(timep):group_voweles_0.i  6.221    7.201 1770.941 < 2e-16 ***
s(timep):group_voweles_1.i  5.245    6.298  468.819 < 2e-16 ***
s(timep):group_voweles.u    1.000    1.000    0.009 0.92287
s(timep):group_voweles_0.u  7.568    7.945  777.603 < 2e-16 ***
s(timep):group_voweles_1.u  2.484    3.086    4.621 0.00269 **
s(timep,subj)                      56.008 534.000    6.025 < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =  0.939   Deviance explained =   94%
GCV = 0.029735  Scale est. = 0.029455  n = 9720
```

Let's extract the model predictions and inspect them.

```
gam_3_preds <- predict_gam(gam_3, exclude_terms = "s(timep,subj)")
gam_3_preds
```

```
# A tibble: 99 x 6
   group_vowel timep      eud      se  lower_ci upper_ci
   <fct>       <dbl>    <dbl>   <dbl>     <dbl>    <dbl>
 1 es.i            1   2.32    0.0155    2.29     2.35
 2 es.u            1   2.02    0.0155    1.99     2.05
 3 es.a            1   1.53    0.00809   1.52     1.55
 4 es_0.i          1   0.0930  0.0473    0.000345 0.186
 5 es_0.u          1   0.154   0.0476    0.0611   0.248
 6 es_0.a          1  -0.00855 0.0410   -0.0888   0.0717
 7 es_1.i          1   1.10    0.0470    1.00     1.19
 8 es_1.u          1   1.45    0.0447    1.36     1.53
 9 es_1.a          1   1.28    0.0410    1.20     1.36
10 es.i          1.8   2.32    0.0139    2.30     2.35
# i 89 more rows
```

The function `predict_gam()` automatically samples 10 equidistant points from numeric variables (like `timep`, based on the default `length_out = 10` argument) and all levels in categorical variables (like `group_vowel`). For each combination of the sampled points and levels, the function returns the value of the outcome (`eud`) and the standard error (`se`). The lower (`lower_ci`) and upper (`upper_ci`) boundaries of the 95% Confidence Interval are also returned.

Furthermore, `predict_gam()` returns an object of class `tidygam` which can be plotted with `plot()` (similarly to how `ggpredict()` from the ggeffects package works, Lüdecke (2018)). However, we might want to do some processing of the predictions before plotting: in particular, we might want to split the `group_vowel` column into the two original variables, `group` and

`vowel`. We can achieve this straight from the function. And we might also want to extract more than 10 time points, to get a smoother curve.

```r
gam_3_preds_2 <- predict_gam(
  gam_3,
  length_out = 25,
  exclude_terms = "s(timep,subj)",
  separate = list(group_vowel = c("group", "vowel"))
)
gam_3_preds_2
```

```
# A tibble: 234 x 7
   group vowel timep      eud      se  lower_ci upper_ci
   <chr> <chr> <dbl>    <dbl>   <dbl>     <dbl>    <dbl>
 1 es    i         1    2.32   0.0155    2.29     2.35
 2 es    u         1    2.02   0.0155    1.99     2.05
 3 es    a         1    1.53   0.00809   1.52     1.55
 4 es_0  i         1    0.0930 0.0473    0.000345 0.186
 5 es_0  u         1    0.154  0.0476    0.0611   0.248
 6 es_0  a         1   -0.00855 0.0410  -0.0888   0.0717
 7 es_1  i         1    1.10   0.0470    1.00     1.19
 8 es_1  u         1    1.45   0.0447    1.36     1.53
 9 es_1  a         1    1.28   0.0410    1.20     1.36
10 es    i      1.32    2.32   0.0148    2.30     2.35
# i 224 more rows
```

The syntax `list(group_vowel = c("group", "vowel"))` instructs the function to split the `group_vowel` column into two columns, `group` and `vowel`. Splitting is done based on the `sep_by` argument, which is . by default. Now we can plot the predictions.

```r
gam_3_preds_2 %>%
  plot(series = "timep", comparison = "group") +
  facet_grid(cols = vars(vowel))
```
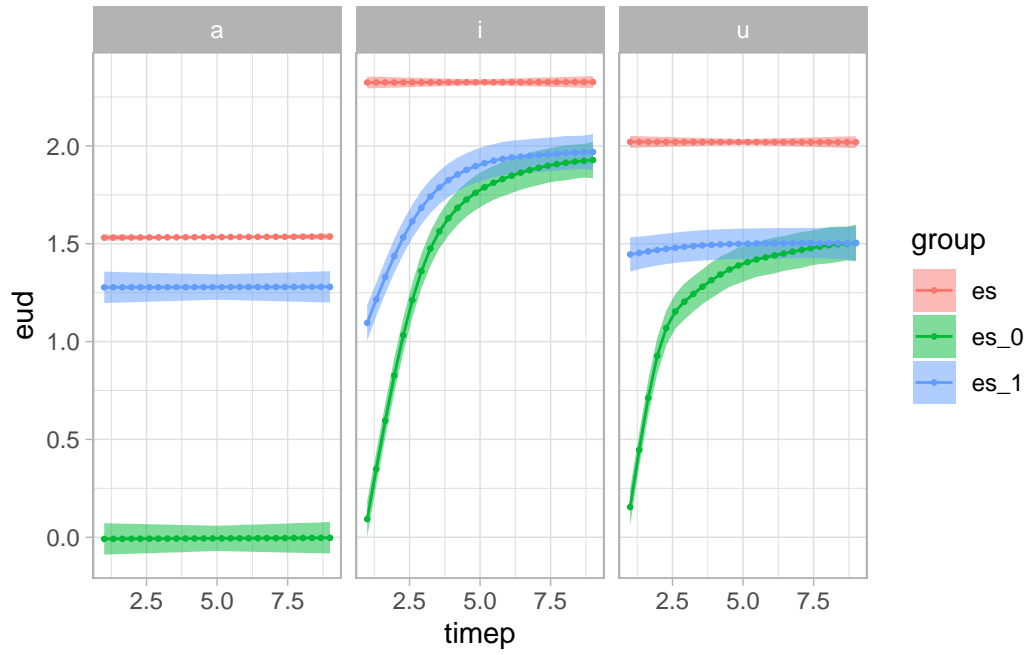
Figure 7: Predicted Euclidean distance along the duration of the vowel, for /a, i, u/ in simultaneous bilinguals (es), beginners late bilinguals (es_0) and advanced late bilinguals (es_1)

References

Aldrich, A. C. (2014). *Acquisition of L2 phonology: An acoustic analysis of the centralization of L2 Spanish /a/ in adult L1 English-speaking learners.* MA Thesis. Brigham Young University.

Bland, J. (2016). *Speech style, syllable stress, and the second-language acquisition of Spanish /e/ and /o/.* MA thesis, Virginia Polytechnic Institute; State University.

Carlucci, L., & Case, J. (2013). On the necessity of U-shaped learning. *Topics in Cognitive Science*, *5*(1), 56–88. https://doi.org/10.1111/tops.12002

Casani, E. (2020). Valutare la competenza morfosintattica in italiano L2. Una validazione corpus-based dei livelli del QCER. In E. Nuzzo, E. Santoro, & I. Vedder (Eds.), *Valutazione e misurazione delle produzioni orali e scritte in italiano lingua seconda* (pp. 15–26). Florence: Franco Cesati Editore.

Cazden, C. B. (1968). The acquisition of noun and verb inflections. *Child Development*, *39*(2), 433–448. https://doi.org/10.2307/1126956

Cobb, K. (2009). *La pronunciación de vocales átonas en español: La aplicación de reglas fonológicas por parte de hablantes no-nativos del español.* MA Thesis. The University of Arizona.

Cobb, K., & Simonet, M. (2015). Adult second language learning of Spanish vowels. *Hispania*, *98*(1), 47–60. https://doi.org/10.1353/hpn.2015.0026

Ervin, S. M., & Miller, W. R. (1963). Language development. In H. W. Stevenson, J. C. Kagan, C. C. Spiker, N. B. Henry, & H. G. Richey (Eds.), *Child psychology: The sixty-second yearbook of the National Society for the Study of Education, Part 1* (pp. 108–143). University of Chicago Press.

Gass, S., & Selinker, L. (2008). *Second language acquisition. An introductory course.* New York: Routledge.

Geeslin, K. L., & Guijarro-Fuentes, P. (2006). A longitudinal study of copula choice: Following development in variable structures. In N. Sagarra & A. J. Toribio (Eds.), *Selected procedures of the 9th hispanic linguistics symposium* (pp. 144–156). Somerville, MA: Cascadilla Proceedings Project.

Hastie, T., & Tibshirani, R. (1986). Generalized additive models. *Statistical Science*, *1*(3), 297–310. https://doi.org/10.1201/9780203753781-6

Iruela, A. (1997). Adquisición del vocalismo español por holandeses: Análisis en estilo semiespontáneo. *Estudios de Fonética Experimental IX*, 135–180.

Kellerman, E. (1985). If at first you do succeed... In S. M. Gass & C. G. Madden (Eds.), *Input in Second Language Acquisition* (pp. 345–353). Rowley: Newbury.

Lightbown, P. (1983). Exploring relationships between developmental and instructional sequences in L2 acquisition. In H. Seliger & M. Long (Eds.), *Classroom oriented research in second language acquisition* (pp. 217–243). Rowley, MA: Newbury House.

Long, M. H. (1990). The least a second language acquisition theory needs to explain. *TESOL Quarterly*, *24*(4), 649–666. https://doi.org/10.2307/3587113

Lüdecke, D. (2018). Ggeffects: Tidy data frames of marginal effects from regression models. *Journal of Open Source Software*, *3*(26), 772. https://doi.org/10.21105/joss.00772

Marcus, G. F., Pinker, S., Ullman, M., Hollander, M., Rosen, T. J., Xu, F., & Clahsen, H. (1992). Overregularization in language acquisition. *Monographs of the Society for Research in Child Development*, *57*(4), i–178. https://doi.org/10.2307/1166115

Martínez Celdrán, E. (1984). *Fonética.* Barcelona: Teide.

McElreath, R. (2019). *Statistical rethinking: A Bayesian course with examples in R and Stan.* Boca Raton, FL: CRC Press.

Pliatsikas, C., & Marinis, T. (2013). Processing empty categories in a second language: When naturalistic exposure fills the (intermediate) gap. *Bilingualism: Language and Cognition*, *16*(1), 167–182. https://doi.org/10.1017/S136672891200017X

R Core Team. (2023). *R: A language and environment for statistical computing.* R Foundation for Statistical Computing. https://www.R-project.org/

Rumelhart, D. E., & McClelland, J. L. (1986). On learning the past tenses of English verbs. Implicit rules or parallel distributed processing? In J. L. McClelland & D. E. Rumelhart (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition.* Cambridge: MIT Press.

Shirai, Y. (1990). U-shaped behavior in L2 acquisition. *Variability in Second Language Acquisition: Proceedings of the Tenth Meeting of the Second Language Research Forum*, *2*, 685–700.

Sóskuthy, M. (2017). *Generalised additive mixed models for dynamic analysis in linguistics: A practical introduction.* arXiv.org preprint, arXiv:1703.05339.

Sóskuthy, M. (2021). Evaluating generalised additive mixed modelling strategies for dynamic speech analysis. *Journal of Phonetics*, *84*, 101017. https://doi.org/10.1016/j.wocn.2020.101017

Tamminga, M., Ahern, C., & Ecay, A. (2016). Generalized additive mixed models for intraspeaker variation. *Linguistics Vanguard*, *2*(s1), 33–41. https://doi.org/10.1515/lingvan-2016-0030

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. https://doi.org/10.1016/j.wocn.2018.03.002

Williams, S., Guijarro-Fuentes, P., & Vulchanova, M. (2022). U-shaped trajectories in an L2 context: Evidence from the acquisition of verb morphology. *Vigo International Journal of Applied Linguistics*, *19*, 223–266. https://doi.org/10.35869/vial.v0i19.3764

Winter, B. (2013). *Linear models and linear mixed effects models in R with linguistic applications.* arXiv:1308.5499. /url%7Bhttps://arxiv.org/abs/1308.5499%7D

Winter, B. (2020). *Statistics for linguists: An introduction using R.* Routledge.

Wood, S. (2006). *Generalized additive models: An introduction with R.* CRC Press.