Journal of Language Evolution, 2016, 7–18
doi: 10.1093/jole/lzv003
How-to

How to analyze linguistic change using mixed models, Growth Curve Analysis and Generalized Additive Modeling

Bodo Winter[1,*] and Martijn Wieling[2]

[1]Cognitive and Information Sciences, University of California, Merced, CA, USA and [2]Center for Language and Cognition Groningen, University of Groningen, Groningen, The Netherlands

*Corresponding author: bodo@bodowinter.com

Abstract

When doing empirical studies in the field of language evolution, change over time is an inherent dimension. This tutorial introduces readers to mixed models, Growth Curve Analysis (GCA) and Generalized Additive Models (GAMs). These approaches are ideal for analyzing nonlinear change over time where there are nested dependencies, such as time points within dyad (in repeated interaction experiments) or time points within chain (in iterated learning experiments). In addition, the tutorial gives recommendations for choices about model fitting. Annotated scripts in the online Supplementary Data provide the reader with R code to serve as a springboard for the reader's own analyses.

Key words: mixed models, mixed-effects regression, growth curve analysis, generalized additive modelling

1. Introduction

Language evolution entails change. That is, differences in the quantity or quality of some linguistic phenomenon or language-related biological phenomenon as a function of time. There are plenty of approaches for the analysis of time series data, and it can be a daunting task to choose an appropriate technique for a given dataset. This tutorial outlines two approaches that are particularly general and flexible, and which can be applied to many different kinds of datasets within language evolution research. Our focus will be on Growth Curve Analysis (GCA) (Mirman et al. 2008; Mirman 2014), a variant of mixed models, and Generalized Additive Models (GAMs) (Hastie and Tibshirani 1986; Wood 2006; Wieling et al. 2014).

We will use these two methods to analyze data from iterated learning experiments, a fruitful approach for studying language evolution (for reviews, see, Scott-Phillips and Kirby 2010; Kirby et al. 2014). Iterated learning describes 'the process by which a behaviour arises in one individual through induction on the basis of observations of behaviour in another individual *who acquired that behavior in the same way*' (Kirby et al. 2014: 28, italics as in original). In iterated learning experiments, the output of one participants' language learning behavior serves as the linguistic input for the next participant. This creates a chain of participants, simulating intergenerational transmission of linguistic structures in accelerated time.

Data from such iterated learning experiments tends to have a specific structure. First, there is a temporal dimension: The first participant in a chain is generation $t = 1$, the second participant who receives the first participant's input forms a second generation, $t = 2$. Then,

there is nesting: a participant only matters insofar as she participates in a chain. For example, participants 1, 2, and 3 might be the first-, second-, and third-generation participants of chain A, whereas participants 4, 5, and 6 might belong to a separate chain. Because there is variation between learners, the artificial language generally evolves in a slightly different fashion in every chain. For example, in Kirby et al. (2008), some chains develop more compositional structure than others, and different chains may use different linguistic forms or collapse different meaning distinctions, ultimately rendering each chain unique.

If we want to make general claims about language evolution with this experimental setup, the 'chain' becomes the target of inferential statistics. That is, we may consider each set of chains in the experiment as a sample from a population of chains that we wish to generalize upon. Any appropriate analysis needs to take the variability associated with these chains into account. In the following, we will do this by including 'chain' as a so-called 'random effect' (see below). Both approaches illustrated in this article, GCA and GAMs, allow the user to include random effects.

The structure of this article is as follows. The next section provides a quick overview of some basic regression concepts and introduces mixed-effects modeling for dealing with interdependent data structures (Section 2). Subsequently, Section 3 extends this approach and introduces GCA, while Section 4 introduces GAMs. Finally, Section 5 reviews some other approaches to analyze time series data adequate for different types of data structures. The online Supplementary Data (S1) provide documented R code to help the researcher apply the methods illustrated in this article to his or her own data. The body of the text is intended to introduce the conceptual side of the analysis presented in this article. The online Supplementary Data contain most of the details about how to execute the analyses in practice (i.e. how to compare models, how to retrieve *p*-values using likelihood ratio tests, etc.). The online Supplementary Data are accessible online with the journal or as a Github repository: https://github.com/bodowinter/change_tutorial_materials

2. From regression to mixed models

Consider a researcher who analyzes data from just a single chain in an iterated learning experiment. Starting with an initially random mapping of word forms to meanings, this particular chain ultimately developed a compositional language, with a systematic one-to-one correspondence between forms and meanings. Thus, 'compositionality' increased over time (for ways of measuring compositionality see e.g. Kirby et al. 2008). This can be modeled in a regression framework by regressing compositionality onto time, or, in other words, by using time as a linear predictor of compositionality. Figure 1 shows simulated iterated learning data for which this analysis approach could be used. The estimated regression equation for the data shown in Figure 1 is 'compositionality score = 0.28 + 0.85 * time'. The number 0.28 represents the intercept, which is where the regression line crosses the *y*-axis (the dashed line), that is where 'time' equals zero. The number 0.85 represents the slope of the effect of 'time'. In this particular simulated dataset, the first generation is coded as 1, and thus the lowest predicted value will be 0.28 + 0.85 * 1 = 1.13. The predicted value for the last time point is 0.28 + 0.85 * 10 = 8.78. The intercept always represents the *predicted* y-value (on the regression line) when the variable 'time' equals 0, even if, like in this case, there is no observed data for that value.

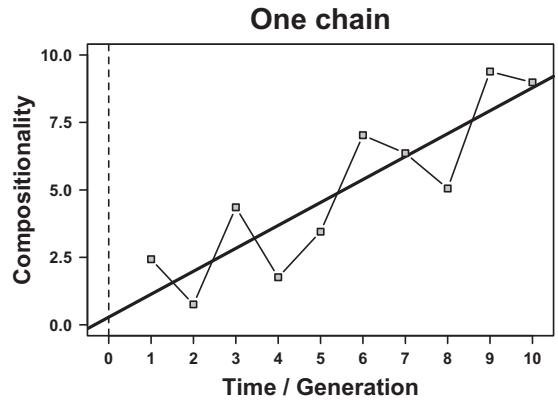

Figure 1. Data from a single chain (gray square dots) with a superimposed regression line indicating the predictions of a linear model (black line). As can be seen, compositionality increases over time.

In the case of just one regression chain where some quantity changes linearly as a function of time, simple linear regression is appropriate. However, once there are multiple chains, it becomes important to statistically account for the differences between chains, that is some chains will be changing more over time, some less. To achieve this, one can use linear mixed-effects regression models (Pinheiro and Bates 2000; Gelman and Hill 2007; Baayen 2008, Ch. 7; Zuur et al. 2009), an extension of regression.

Within the mixed model framework, a critical distinction is made between 'fixed' and 'random' effects (cf. Mirman 2014; see Gelman and Hill 2007, for an alternative conceptual framework not relying on this distinction). Fixed effects can be continuous (such as 'time'

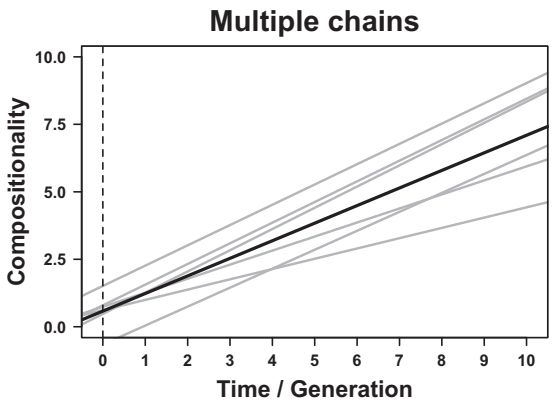

Figure 2. Predictions from multiple chains analyzed using linear mixed-effects regression. The black line shows the overall model fit (the predicted average across chains), the gray lines represent the chain-specific effects obtained by adding intercept and slope adjustments (modeled by the random intercept and random slope) to the overall model fit.

or 'generation') or categorical (such as a condition difference). The prototypical fixed effect is repeatable (i.e. we could design another experiment with exactly those fixed effects) and is expected to have a systematic or predictable influence on the dependent variable (i.e. we expect a similar effect if we were to conduct another study with the same experimental manipulation). In contrast, the prototypical random effect is sampled randomly from a population and is expected to exert idiosyncratic and unpredictable influence on the dependent variable. Random-effect factors are always categorical (e.g. nominal 'subject' or 'chain' identifiers) (see, e.g. Crawley 2010, Ch. 19).

There are two types of random effects, namely, random intercepts and random slopes. Models that include random intercepts are also sometimes called varying-intercept models; models that include random slopes on top of random intercepts are sometimes called varying-slope models (varying-slope models generally also have varying-intercepts). The random intercepts account for variability in the intercept for each level of the random-effect factor. For example, different chains may have different starting levels of compositionality. In contrast, a random slope models variability in the effect of a certain predictor on the dependent variable for each level of the random-effect factor. For example, different chains may vary in how fast compositionality increases over time.

Take, for example, Figure 2, which is simulated data for an iterated learning experiment with six different chains. The black line represents the overall estimate across chains. The gray lines represent the random-effect estimates for every individual chain, obtained by adding the random intercept and random slope for time of each individual chain to the overall intercept and time slope. As can be seen, some chains vary mostly in their intercept, that is, the line is mostly shifted upward or downward. This is modeled by the random intercept component of the model. On the other hand, some chains also vary in their slope, that is, they develop compositional structure more or less quickly than the average chain. This is accounted for by the random slope component of the model.

Within the model, all random intercepts correspond to just a single parameter, and all random slopes correspond to just one more parameter. In each case, the parameter models the variance around the general estimate for all levels of the random effect factor. This variation is assumed to be normally distributed. That is, we expect that many chains have intercepts that deviate only minimally from the grand intercept (of the black line in Figure 2), and we expect some chains to have very different intercepts. Likewise, we assume the slopes to be normally distributed around the mean slope,[1] with some slopes deviating more strongly from the overall estimate than others.

A feature of mixed models is that random-effect estimates (in both intercepts and slopes) are drawn toward the fixed-effects prediction, a phenomenon called *shrinkage* (Pinheiro and Bates 2000: Ch. 4; Baayen 2008: 275–278; cf. discussion in Mirman et al. 2012). If one were to run individual regression analyses for each chain, there would be no shrinkage (cf. Gellman and Hill 2007: Ch. 11–13). This means that the individual intercepts and slopes would be farther away from the predicted grand mean than the gray lines shown in Figure 2. From a sampling perspective, shrinkage can be likened to the phenomenon of 'regression towards the mean' (Bland and Altman 1994; Kahneman 2011: Ch. 17): Across the board, if deviations from a mean are normally distributed, we expect extreme deviations to be less extreme the next time we measure something. Take, for example, the chain with the very shallow slope in Figure 2: given that this line is extracted from a mixed-effects regression model, it has been adjusted to be more closely positioned to the grand mean

1 The fact that individual random effects estimates are visualized in Figure 1b does not contradict the statement that only one parameter is fitted for each random effect. The random-effects estimates for each chain are *posterior* estimates (so-called Best Linear Unbiased Predictors, or BLUPs), derived from the fitted model. These estimates exhibit shrinkage toward the mean.

Table 1. Common representations of mixed models for the example of an iterated learning model with a random effect of chain. (The mathematical notations focus on the fixed effect coefficients and omit error terms, i.e. we focus on the expectation rather than the outcome; the last two formulas are the same because they are both for models with varying intercepts and varying slopes)

	Simplified mathematical notation	R syntax (package lme4)
Random intercept only	$y_i = \alpha_i + \beta t$	y ~ time + (1|chain)
Random intercepts and (uncorrelated) slopes	$y_i = \alpha_i + \beta_i t$	y ~ time + (1|chain) + (0+time|chain)
Random intercept and slopes (with correlation)	$y_i = \alpha_i + \beta_i t$	y ~ time + (1+time|chain)

(represented by the black line) than if we would have fitted a separate linear regression model for this specific chain. The idea behind this is that if we were to do the same experiment again with the same participants, the slope would likely be less extreme than in the previous experiment. Because of this, shrinkage of the random-effects estimates is often a desired aspect of mixed models because it allows for a more accurate depiction of individual differences (in this case, differences between chains, in other cases, differences between participants, see Mirman Yee et al. 2011).

Table 1 shows a simplified mathematical representation of mixed models together with common R syntax used in the lme4 package (Bates et al. 2014).

The dependent measure y, is modeled as a function of an intercept, α, and as a function of time t. The slope β is multiplied with each time value. These aspects of the equation correspond to a simple regression equation, $y = \alpha + \beta t$. The sub-indices i shown in Table 1 are a simplified way of representing the functionality of a mixed-effects regression model. In this case, these indices i correspond to the levels of the random-effect factor. For example, $i = 1$ is the first chain, while $i = 2$ is the second chain. Thus, this simplified mathematical notation embodies the insight that both the intercept and the slope may vary between chains.

The lme4 syntax in Table 1 specifies y as a function of time (the fixed effect). The parts in brackets represent the random effects, where '(1|chain)' represents random intercepts for chain. The notation can be explained as follows: '1' stands for the intercept,[2] which is conditioned on chain by the vertical bar. The expression '(0+time|chain)' means that the slope of the time effect is conditioned on chain as well. The '0+' part indicates that random intercepts are not fit again, thus the term '(0+time|chain)' *only* adds a random slope. Alternatively, one can specify intercepts and slopes together with '(1+time|chain)', where both the intercept and the slope of time are conditioned on chain. The difference between these two alternative ways of specifying random slopes is the following: if both intercept and slope are conditioned in one term, as in '(1+time|chain)', the model additionally estimates a random intercept/slope correlation. In substantive terms, such a correlation could exist if chains that start low in compositionality (they have small intercepts) increase compositionality more quickly (they also have steeper slopes), in which case intercepts and slopes are positively correlated. To suppress estimating such a correlation, the 'de-correlated' random-effects specification '(1|chain) + (0+time|chain)' can be used. Because including slope/intercept correlations may lead to overly complex models, their inclusion should be assessed via model comparison (see also below).[3]

Random slopes turn out to be crucial and warrant special attention. A random slope is always linked *to some fixed effect*. Figure 2 visualizes the by-chain random slope of the fixed effect predictor time. This means that the effect of time is allowed to vary for each chain. Failing to fit a random slope for the time effect amounts to assuming that every chain changes at exactly the same rate. This is almost never a feasible assumption to make, especially since past research in the iterated learning paradigm shows that chains do in fact differ in how they change (e.g. Kirby et al. 2008). It has been recommended to fit random slopes for critical variables of interest (Barr et al. 2013), because the significance of fixed-effects estimates may be anti-conservative (i.e. yielding Type I errors, spuriously significant results) when the corresponding random slope is not in the model (Schielzeth and Forstmeier 2009). However, a complex random-effects structure may result in overfitting and failure to converge to stable estimates[4]

2 The fact that the intercept is represented by a '1' in the R syntax has to do with the underlying linear algebra of these models, where mathematically the intercept in the model matrix is represented by a column of 1's.

3 In their simulation study, Barr et al. (2013) show that Type I error rates are not strongly affected depending on whether one does or does not estimate random intercept/random slope correlations.

4 An explanation about convergence is in order here: whereas for simple linear regression there is an analytical solution to derive the best-fitting regression model

(Bates et al. 2015). At present, we suggest evaluating which random-effects structure is supported by the data by using likelihood ratio tests (Bates et al. 2015), as demonstrated in the online Supplementary Data.

For a short general introduction to linear models and mixed models, see Winter (2013). For a more comprehensive introduction to mixed models, we recommend Gellman and Hill (2007) and Baayen (2008, Ch. 7). For discussion of the random-effects structure of mixed models, see Barr et al. (2013) and Bates et al. (2015).

3. Mixed models with polynomial predictors: GCA

So far, we have assumed that the relationship between the dependent measure *y* and time is linear. If visualization of the temporal trajectories reveals nonlinearities, more complex models need to be considered. Luckily, there are several ways to use mixed models and extensions of mixed models to deal with nonlinear data. This section deals with what is sometimes called GCA (e.g. Mirman et al. 2008; Mirman 2011). Section 4 discusses a more flexible alternative, GAMs.

Consider an experiment where participants play a game of 'vocal charades', as in the study of Perlman et al. (2015). At each round, a participant has to vocalize a meaning to the partner (e.g. 'ugly') without using language (e.g. through grunting or hissing). The partner has to guess the meaning of the vocalization. This game is played repeatedly with the finding that over time, a dyad converges on a set of nonlinguistic vocalizations that assure a high degree of intelligibility between the two participants in the dyad (Perlman et al. 2015). Initially, participants may be struggling with the task and explore very different kinds of vocalizations. Over time, they may converge on a more stable set of iconic vocalizations, that is vocalizations that resemble the intended referent (e.g. a high-pitched sound for 'attractive' and a low-pitched sound for 'ugly').

(i.e. a set of mathematical operations that need to be applied to the data), the parameters of more complex models such as mixed models need to be estimated numerically. This process is an algorithmic search process, with the goal of finding the parameter values of the model that have the highest likelihood. In some cases, in particular when the fitted model is complex and the data too sparse to support this complexity, the algorithm will not yield a stable end result. Bates et al. (2015) and Jaeger et al. (2011) discuss convergence issues and issues surrounding the estimation of complex random-effects structures in more detail.

Finally, after even more time, the dyad may conventionalize to idiosyncratic patterns that deviate from iconicity and become increasingly arbitrary (cf. Garrod et al. 2007). This general pattern is shown with simulated data in Figure 3a, with iconicity first increasing, and then decreasing slightly, as signals become more and more arbitrary through conventionalization.[5] Such an inverse U-shaped pattern can be modeled by incorporating polynomial fixed effects into the mixed-effects regression analysis. This approach is frequently called GCA in psychology (Mirman et al. 2008; Mirman 2014).

Figure 3b shows the predictions of the GCA for a simple model in which the dependent variable 'iconicity' is modeled as a function of interaction round entered both as a linear predictor (untransformed: dubbed 'time') and as a quadratic predictor ('time2', obtained by squaring 'time'). Each predictor captures a different aspect of the trajectory of iconicity over time. The linear predictor captures the overall increase or decrease over time. The quadratic predictor captures how much the curve is bent upwards or downwards.

Both linear and quadratic effects are associated with their own slopes. In this case, the quadratic effect turns out to be negative (−0.22 for the dataset shown in Figure 3), which corresponds to the inverse U-shape (a regular U-shape would correspond to a positive quadratic effect). The linear effect for this dataset is positive (+2.83), corresponding to the fact that even though there is a strong quadratic effect, iconicity at the end of the experiment is higher than at the beginning. Consequently, the inverse U-shape is slightly tilted upwards.

One way to model this data with a mixed-effects regression model is specified in Equation (1) (in a simplified fashion, focusing on the expected value). The corresponding R lme4 syntax is given in (2). The result of running the mixed model is given in (3), with the estimated coefficients:

$$iconicity_i = \alpha_i + \beta_{1i}t + \beta_{2i}t^2 \quad (1)$$

$$\begin{aligned}\text{lmer}(\text{iconicity} \sim \text{time} + \text{time2} + \ & (1|\text{dyad}) \\ + (0 + \text{time}|\text{dyad}) \ + \ & (0 + \text{time2}|\text{dyad}))\end{aligned} \quad (2)$$

$$iconicity_i = -0.32_i + 2.83_i t + -0.22_i t^2 \quad (3)$$

Notice that there are now two slopes, one for the effect of linear time (β_1) and one for the effect of quadratic

5 Perlman et al. (2015) did not find a completely inverse U-shape curve but instead accuracy and iconicity plateaued out.

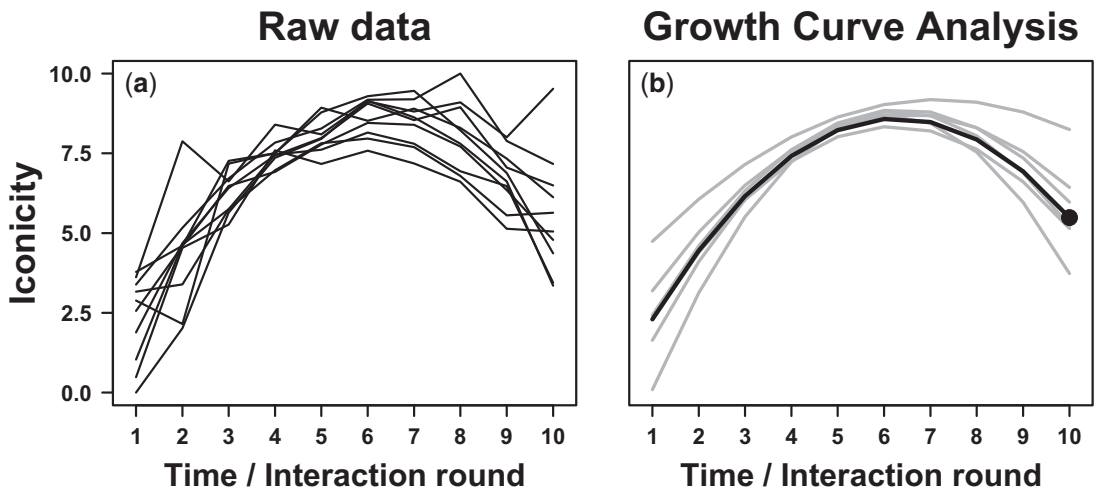

Figure 3. (a) Raw results from a simulated repeated interaction experiment (inspired by Perlman et al. 2015). The *x*-axis represents the sequential rounds for which the vocal charades game was played, the *y*-axis represents an iconicity measure. Each line represents one dyad. (b) The predictions for the fixed effect (black line) and the random effects per dyad (gray lines) of a growth curve model. The black dot represents a particular point discussed in the body of the text.

time (β_2), both of which are allowed to differ by dyad, as indicated by the *i* indices. The formulation in (2) assumes that the random intercept and slopes are all uncorrelated, in contrast to the more complex correlated random-effects structure '(1+time+time2|dyad)'. For the model visualized in Figure 3b, three random effects were fitted. First, random intercepts for each dyad, represented by '(1|dyad)' (each line starts at a slightly different iconicity value at the intercept). Second, random slopes for the linear effect, represented by '(0+time|dyad)' (the overall increase or decrease of iconicity is slightly different for each dyad). Third, random slopes for the quadratic effect, represented by '(0+time2|dyad)' (the degree of bending is different for each dyad). All of these random effects are needed to allow the entire curve to vary for each dyad. Whether this additional freedom (i.e. allowing the curve to vary by dyad, etc.) is necessary can be tested using a likelihood ratio test. In particular, we recommend fitting a model without the correlation parameter and a model with the correlation parameter. If a likelihood ratio test comparing these two models is significant, this means that the added parameter is supported by the data. If the test is not significant, the additional correlation parameter can be dropped, as there is no statistically detectable difference between the two models (i.e. the simpler model is preferred).

To understand what the model predicts, consider the point $t = 10$, represented by the black dot in Figure 3b. The predicted value (i.e. the fitted value) for this point can be derived by inserting '10' into the regression equation shown in (3): $iconicity_i = -0.32 + 2.83 * 10 + -0.22 * 10^2$. This yields the value 5.98, the predicted iconicity for round 10 of the vocal charades game. Crucially, the time value has to be squared ($10^2 = 100$) before being multiplied with the coefficient for the quadratic term.

To get estimates for the particular dyads, the slope adjustments for the particular dyads need to be incorporated (see online Supplementary Data S1). Even though the model accounts for a nonlinear (quadratic) pattern, this is still a *linear* mixed-effects model. That is because iconicity is modeled as the linear combination of the two fixed effects (time and time2). However, because one of the two fixed effects is a quadratic transformation of the other, the model is able to account for the quadratic pattern in the data.

It is possible to fit more complicated curves. For example, one could add the time variable as a cubic predictor, 'time3'. Even higher order polynomials are possible (to the power of four, five, ...). By incorporating more and more polynomial transformations of the time variable, increasingly complex curves can be modeled with increasingly more fidelity. What order of polynomial is needed for a given dataset? A common approach is to use likelihood ratio tests to first establish what order of polynomials is needed (e.g. up to cubic), then assess the influence of the predictors of interest within that polynomial structure. This is demonstrated in the online Supplementary Data (S1), where it is

shown that for the data in Figure 3a, a model with a cubic term is not significantly different from a model with a quadratic term ($\chi^2(1) = 0.1264$, $P = 0.72$). That is, a cubic model does not significantly improve the fit over and above a quadratic model, and therefore the simpler model is chosen. Then, within this quadratic model, the time effect is assessed.

In general, it is desirable to keep the number of polynomial transformations low. Increasingly more complex models become increasingly more difficult to interpret. And, on the practical side, fitting more parameters means that the model estimation process becomes more difficult, which may lead to convergence failures. In the case of the dataset shown in Figure 3, one might argue that on top of the quadratic shape being apparent in the plot, fitting a quadratic effect may be theoretically motivated. This is because past research has shown that iconicity decreases as patterns become conventionalized (Garrod et al. 2007). Hence, we expect an increase of iconicity (as dyads become less random) followed by a slight decrease (due to conventionalization), or at least a plateauing out of iconicity (as observed in Perlman et al. 2015). Consequently, fitting a quadratic effect is theoretically justifiable in this case.

To aid the interpretation of Growth Curve models, it generally makes sense to center each predictor, that is, subtracting the mean of time from time, and subtracting the mean of time2 from time2, and so on. This sets the intercept to the mean of the time series. For a discussion of the interpretational benefits of centering, see Schielzeth (2010). Centering does not change the nature of the model since it is a linear transformation of the data and the relationship between values is maintained. Centering may also prevent spurious correlations of random intercepts and slopes (Baayen 2008, Ch. 7). Thus, we recommend the motto 'if in doubt, center' as a best practice for most if not all analyses based on regression.

For accessible introductions to GCA (including a useful review of mixed-effects regression models), see Mirman (2014).

4. Limits of polynomials: motivating GAMs

Polynomials can model many curves and will be sufficient for many datasets that come up in language evolution experiments. Polynomials, do, however, have limitations. In the case of the GCA discussed in the previous section, the curve is constrained to be a combination of a linear term and a quadratic term. In particular, polynomials tend to have problems with long asymptotes and plateaus (Figure 4a) or curves that have too many bends (Figure 4b). The dashed lines show the relatively badly fitting predictions of a simple polynomial regression model. The bold lines show the predictions of a GAM, which captures the behavior of the data more adequately in these cases.

GAMs, originally developed by Hastie and Tibshirani (1986), relax many of the restrictions of GCA. In the particular case displayed in Figure 4, the time variable was entered as a thin plate regression spline (TPRS; Wood 2003), which means that a smoothed function is fitted by combining several low-level functions (such as a linear function, a quadratic function, a logarithmic function, etc.) across the whole time span. There are other types of smooths, but thin plate regression splines generally yield the best performance in terms of mean squared error. The appropriate degree of nonlinearity, or 'wiggliness', of the curve is determined on the basis of cross-validation (we will not deal with cross-validation here, but see James et al. 2014 for a general introduction of data mining concepts).

Within the generalized additive modeling framework, both random intercepts and random (linear) slopes can be included. Furthermore, with so-called factor smooths, random variability in nonlinear patterns may be modeled. Conceptually, these factor smooths correspond to random slopes and intercepts. Just as random slopes allow the linear or polynomial lines to differ between chains or dyads, factor smooths allow complex nonlinear trajectories to differ between chains or dyads. As factor smooths can vary in height for each dyad, they also encapsulate random intercepts. And, just as with mixed models, where it is important to consider random slopes and intercepts to keep Type I error rates low, factor smooths are important to ensure that the estimated nonlinear trajectories have conservative confidence bands. If, for example, a complex nonlinear trajectory through time were largely driven by just a single dyad, this would be missed without adding a factor smooth over time per dyad. Thus, omitting factor smooths amounts to assuming that all dyads (or chains) behave exactly the same way with respect to the time variable.

The R syntax using the function bam from the mgcv package (Wood 2006) in (4) shows how to analyze the quadratic data shown in Figure 3a with a GAM:

$$\begin{aligned}\text{bam}(\text{iconicity} \sim\ & \text{s}(\text{time}, \text{k} = 5) \\ & + \text{s}(\text{time}, \text{dyad}, \text{k} = 5, \text{bs} = \text{'fs'}, \text{m} = 1))\end{aligned} \tag{4}$$

This particular GAM estimates a potentially nonlinear effect of time, represented by the term 's(time, k=5)'.

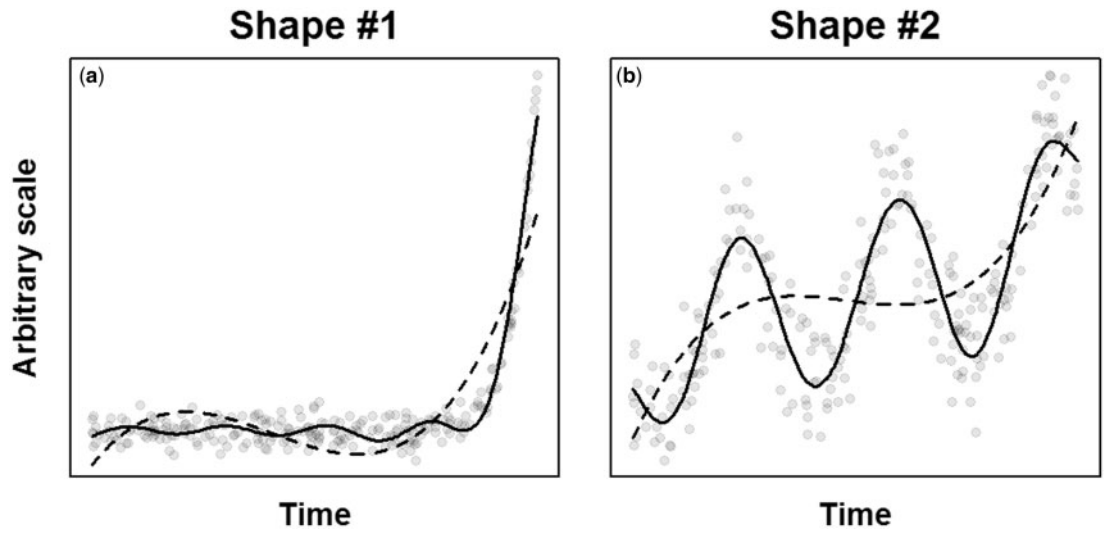

Figure 4. Two curves that create difficulties for GCA, an (a) exponential curve and (b) a sinusoidal pattern. The dashed lines represent growth curve models with polynomials up to the fourth degree ($y = t + t^2 + t^3 + t^4$), the solid lines represent GAM fits.

The 'k' parameter limits the number of low-level functions used to construct the thin plate regression spline. As a rule of thumb, this value should be set to at most half the number of unique time points (in our case 10, so the maximum is set to 5), in order to prevent fitting every individual data point (this rule of thumb is based on the Nyquist frequency). Since this nonlinear pattern varies between dyads, factor smooths need to be fitted to take this variation into account. This is done via the term 's(time, dyad, k=5, bs='fs', m=1)'. The argument 'bs' stands for 'smoothing basis' and 'fs' indicates that a factor smooth is used here. The parameter 'm' controls the degree of smoothness, and is reduced from its default (m = 2) as random effects should not fit the observed patterns perfectly, but allow for shrinkage toward the mean (see Section 2).

Note that separate random intercepts and slopes are not necessary when including factor smooths, as the factor smooths are estimated to have different heights (intercepts) and different slopes (inherent in the nonlinear approach which estimates the whole trajectory). However, when only linear patterns are observed their use is appropriate. In the mgcv package, the specification of a random intercept per dyad is 's(dyad, bs='re')', with bs='re' indicating a random effect. Similarly, a (linear) random slope for time per dyad would be specified by 's(dyad,time,bs='re')'. These terms directly correspond to the lme4 syntax of '(1|dyad)' and '(0+time|dyad)' discussed above. Note that the order of the terms is different from that of the factor smooths. When using bs = 're', the first parameter is always the random-effect factor (in this case, dyad), whereas in the case of factor smooths, it is the second parameter.

In contrast to GCA, GAMs do not have straightforwardly interpretable coefficients. Each smooth is associated with a *p*-value, indicating if it is significantly different from 0, and an 'edf' value (effective degrees of freedom), indicating the degree of nonlinearity (edf = 1 corresponds to a linear pattern, edf > 1 corresponds to a nonlinear pattern). Because the model summary does not include easily interpretable coefficients, visualization of model fits is *essential*. GAM fits for the iconicity data discussed above are shown in Figure 5a. Figure 5b shows the random smooth terms for each dyad, which represents the *difference* of each dyad with respect to the overall trajectory.

An advantage of GAMs over GCA is that the common implementation used in R (the mgcv package, Wood 2006) allows one to account for autocorrelation in the residuals of the model, that is, mutual dependence of consecutive time points. If this autocorrelation is present and not corrected for, one of the assumptions of the regression model (i.e. independence of the observations) is violated, with an associated increase in the rate of Type I errors. The presence of this type of violation therefore needs to be investigated. For this purpose, the R function acf (from the base package) can be used. Note that the data need to be sorted per separate time series in order for this function to work. Figure 6 shows that there is no significant autocorrelation present in the residuals at lag 1 (i.e. residuals at time point t are not correlated with time points at $t + 1$). The reason for this

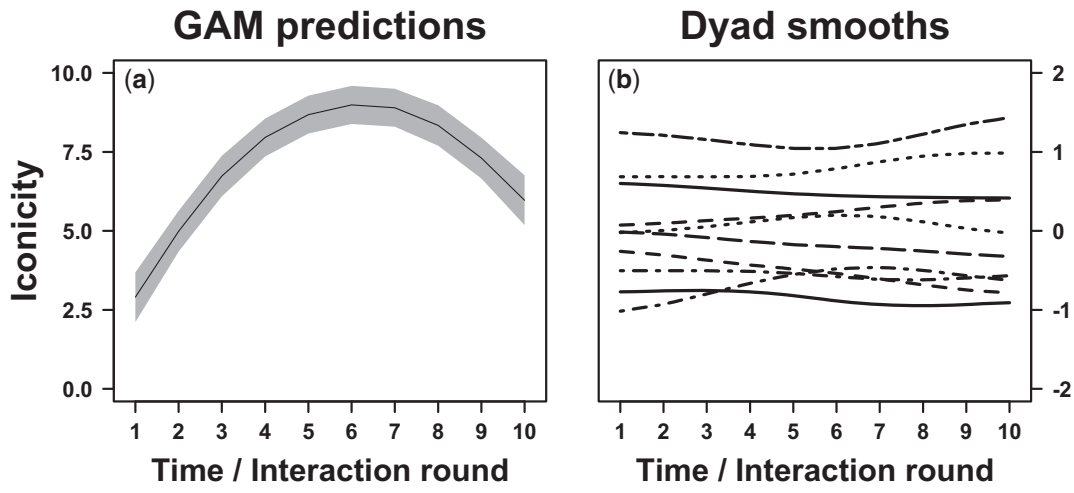

Figure 5. (a) GAM fit and 95% confidence interval of the iconicity data (also shown in Figure 3). (b) Random smooth terms ('factor smooths') that represent the by-dyad differences of the time effect.

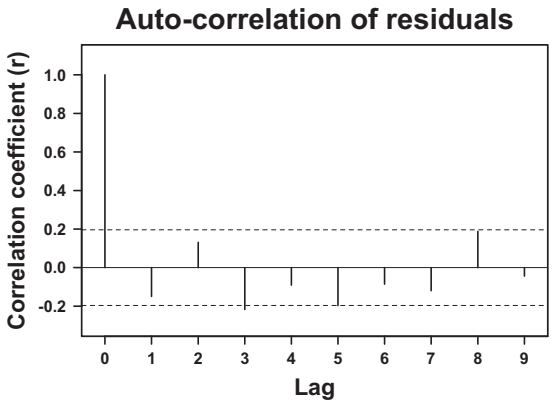

Figure 6. (a) Autocorrelation of the residuals of model (4). The first line is always equal to 1 (residuals at time t are exactly correlated with residuals at time t), the second line is the line of interest and indicates the autocorrelation of the residuals at lag 1 (t versus $t+1$). In this case, the autocorrelation at lag 1 is not significant, as indicated by the fact that the second line is within the area circumscribed by the dashed lines.

is that our model provides a very good fit to the data. Not only the general pattern is fitted, but also how each dyad-specific pattern deviates from this general pattern. When the dataset would be much larger, with for example many separate patterns per dyad, autocorrelation in the residuals is more likely to be present. The online Supplementary Data S1 show how two parameters of the function bam (rho and AR.start) can be used to correct for autocorrelation, if present.

The GAM approach is inherently more flexible than GCA. GAMs can model *any* type of nonlinearity and will adequately take into account individual variation and autocorrelation in the residuals. Even if the general pattern resembles a polynomial, GCA is only appropriate when the individual patterns (the random effects) can also be adequately represented with the same type of polynomial. In many cases, this does not hold.

If, however, the outcome of a particular iterated learning or dyadic interaction experiment looks linear upon visualizing the main trends, and if a quadratic or nonlinear model is not supported by the data, then proceeding with a simple linear mixed-effects regression model with a linear effect of time is a parsimonious approach that is easily interpretable by a general audience. Furthermore, if the polynomials have a specific theoretical significance (as in the iconicity example above) and their coefficients need to be interpreted and reported, GCA may be preferable. Finally, it should be noted that GAMs provide a powerful and flexible approach for dealing with more complex shapes not considered in this article, such as two-dimensional nonlinear interactions between predictors (see Wieling et al. 2014 for an application of GAMs to model the influence of spatial coordinates on lexical variation).

Accessible introductions to GAMs are presented by Clark (2012) and Zuur et al. (2009, Ch. 3). For discussions of GAMs in linguistic contexts, see De Cat et al. (2015), Meulman et al. 2015, Van Rij et al. (in press), and Wieling et al. (2011, 2014).

5. Extensions and alternative approaches

In this section, we discuss several extensions and alternative approaches intended to give the reader a brief

overview of the landscape of methods that can be used to analyze data involving change over time.

So far we have only dealt with continuous dependent measures. What if the dependent measures are not continuous, but categorical? Here, two types of categorical data structures are of particular importance. The first are binary differences, such as 'correct' versus 'incorrect' or 'regular past tense' versus 'irregular past tense'. In this case, a *logistic regression* model should be used. This approach, too, is subsumed by mixed-effects regression models, GCA and GAMs, and can be implemented by simply specifying family='binomial' in the respective model specifications.

When one fits a logistic regression model, care needs to be taken in interpreting the model since all coefficients are represented in the form of logits (the log of the odds of observing one alternative versus the other). Other than that, everything discussed so far carries over to the case of logistic models, with the exception of correcting for autocorrelation in the residuals by the GAMs. Given that a logistic regression model does not have residuals in the same sense as a normal mixed-effects regression model has, correcting for autocorrelation with logistic GAMs is not possible.

For an excellent discussion of logistic mixed-effects regression models, see Jaeger (2008). For a comprehensive and accessible introduction to logistic regression, including logistic mixed-effects regression models, see Gellman and Hill (2007). Logistic regression models are preferable over analyzing percentages or proportions with linear models, as the latter approach may result in proportions that are estimated to be lower than 0 and greater than 1, and it may also result in an increase in the rate of Type I errors (Jaeger 2008). Unfortunately, analyzing percentages or proportions with simple (nonlogistic) linear models is still common practice (see discussion in Jaeger 2008), including in language evolution research.

If the dependent measure is not a binary categorical variable, but a count variable (e.g. counts of words, counts of letters), a *Poisson model* is preferred. Again, as with logistic models, GCA and GAMs can readily incorporate this approach by specifying family='poisson' in the R model specification. Similar to the case of logistic models, most things discussed so far carry over to Poisson models, except that the coefficients are now logged values. While some researchers simply transform count data by taking the log of the counts, or they compute rates (e.g. rate of errors over time), it is advisable to consider a Poisson model. One of the reasons for this is that these models are better suited for dealing with heteroskedasticity (which is beyond the scope of this article, but for a discussion of the issues involved, see O'Hara and Kotze 2010). Zuur et al. (2009) and Coxe et al. (2009) provide accessible introductions to Poisson models.

As is clear from this discussion, one of the advantages of GCA and GAMs is that they can readily be extended to noncontinuous dependent measures, such as binary categorical data (logistic models) or categorical count data (Poisson models). This is because both GCA and GAM are extensions of the *generalized linear model framework*, where the 'generalized' stands for the ability of these models to incorporate error structures that are not assumed to be normal, as is necessary for dealing with categorical data. Part of the flexibility of the approaches discussed in this article stems from this 'generalized' aspect of GCA and GAMs.

The examples discussed above are admittedly simple, intended to focus on the main issue of dealing with change over time and nonlinearities in change over time. When conducting data analysis, many decisions have to be made about which terms to include and which terms to leave out. A particularly important aspect of model fitting to discuss is the presence of interactions. In many cases, researchers will want to include the interaction of time and a condition variable. For example, half of the chains may be seeded with one set of words, the other half with another set of words. Accounting for differences over time in these two conditions is easily done with both GCA and GAMs, as shown by the example syntax in (5) and (6) (we only show the fixed-effect component).

$$\begin{aligned} y \sim\ & \text{time} + \text{time2} + \text{condition} + \text{time} \\ & : \text{condition} + \text{time2} : \text{condition} \end{aligned} \tag{5}$$

$$y \sim s(\text{time}, \text{by} = \text{condition}) + \text{condition} \tag{6}$$

The GCA in (5) fits a linear and quadratic time effect, as well as a condition effect. In addition to this, the interaction of linear and quadratic time with condition is estimated. If condition is a categorical variable (e.g. condition A versus condition B), then the term 'time:condition' indicates how much the linear time slope differs for one set of chains (condition B) versus another set of chains (condition A). Similarly, the term 'time2:condition' indicates how much the quadratic bend of a trajectory over time differs between the two conditions.

The GAM in (6) fits the nonlinear pattern over time for both conditions using the by-parameter. The constant difference between the two conditions is captured by the main effect of condition. To visually assess the difference over time between the two patterns, the function plot_diff from the itsadug R package (van Rij et al. 2015) can be used after fitting the model.

In general, it is advisable to make decisions about which terms should or should not be included in the

model as much as possible based on theory and knowledge about the phenomenon under study. That is, questions such as the following should guide modeling decisions: 'Do I expect chains to differ by condition?' (If yes, add a condition effect.), 'Will the conditions differ with respect to how the effect unfolds over time?' (If yes, add an interaction between time and condition.).

The researcher's model formulation corresponds to the set of theories and beliefs that a researcher has about a data set. The same reasoning extends to more complex cases, where the researcher asks similar questions with respect to additional variables that could matter (such as age, gender, etc.). Basing the model formulation process as much as possible on theory and established knowledge avoids fishing expeditions (searching for significant effects) and uncovering spurious relationships. Of course, when this theory-based analysis leads to a confirmation (or a refutation) of one's hypothesis, a subsequent exploratory analysis may yield additional insights with respect to one's assumptions. For example, a hypothesized effect may not be observable, unless one controls for word frequency. However, it is important to clearly separate one's confirmatory analyses (e.g. testing for a predicted effect of condition differences) from one's exploratory analyses.

Perhaps even more important than the final decision of which model to fit, is transparency about the data analysis process. In any given data analysis, there are myriad decisions to be made (such as decisions about which predictors to include). Only reporting the final model obscures many of these decisions. Therefore, we advocate that researchers publish their full script and, if possible, the data together with their publication. For example, The Mind Research Repository[6] is an excellent example of a repository containing the data and methods associated with a large number of publications (mainly) in linguistics. Sharing data and methods will help foster openness in the language evolution research community, and it builds a stronger knowledge base, where other people can replicate analyses and beginning researchers can learn from published data analyses. While this applies to science in general, it is perhaps even more important in an interdisciplinary field such as language evolution research, as researchers have vastly different backgrounds, including different backgrounds with respect to statistics. For a burgeoning enterprise such as the field of language evolution, transparency of data analyses and willingness to share knowledge is crucial.

6 http://openscience.uni-leipzig.de/index.php/mr2

6. Conclusions

Since language evolution is inherently about change over time, statistical methods are necessary which are able to adequately analyze change. Here, three approaches were introduced, mixed models, GCA and GAM, using simulated iterated learning and repeated interaction experimental results as examples. We demonstrated all approaches with a simple example, outlining pathways for more complicated analyses that suit the individual researcher's need. The generalized (mixed-effects) regression framework, including GCA and GAMs, provides an extremely useful set of tools for analyzing data within the field of language evolution.

Supplementary data

Supplementary data is available at *Journal of Language Evolution* online.

References

Baayen, R. H. (2008) *Analyzing Linguistic Data: A Practical Introduction to Statistics Using R*. Cambridge: Cambridge University Press.

Barr, D. J. et al. (2013) 'Random Effects Structure for Confirmatory Hypothesis Testing: Keep it Maximal', *Journal of Memory and Language*, 68: 255–78.

Bates, D. et al. (2014) *lme4: Linear Mixed-Effects Models using Eigen and S4*. R package version 1.1–7.

—— (2015) 'Parsimonious Mixed Models', *arXiv*:1506.04967.

Bland, J. M., and Altman, D. G. (1994) 'Statistics Notes: Some Examples of Regression Towards the Mean', *British Medical Journal*, 309: 780.

Clark, M. (2012) *Generalized Additive Models: Getting Started with Additive Models in R*. <http://www3.nd.edu/~mclark19/learn/GAMS.pdf> accessed 7 June 2015.

Coxe, S. et al. (2009) 'The Analysis of Count Data: A Gentle Introduction to Poisson Regression and its Alternatives', *Journal of Personality Assessment*, 91: 121–36.

Crawley, M. J. (2013) *The R Book*, 2nd edn. Chichester: John Wiley & Sons.

De Cat, C. et al. (2015) 'Representational Deficit or Processing Effect? An Electrophysiological Study of Noun-Noun Compound Processing by very Advanced L2 Speakers of English', *Frontiers in Psychology*, 6: 77.

Garrod, S. et al. (2007) 'Foundations of Representation: Where Might Graphical Symbol Systems Come From?' *Cognitive Science*, 31: 961–87.

Gelman, A., and Hill, J. (2007) *Data Analysis using Regression and Multilevel/Hierarchical Models*. Cambridge: Cambridge University Press.

Hastie, T., and Tibshirani, R. (1986) 'Generalized Additive Models', *Statistical Science*, 1: 297–310.

Jaeger, T. F. et al. (2011) 'Mixed Effect Models for Genetic and Areal Dependencies in Linguistic Typology', *Linguistic Typology*, 15/2: 281–320.

James, G. et al. (2014) *An Introduction to Statistical Learning: With Applications in R*. New York: Springer.

Kahneman, D. (2011) *Thinking, Fast and Slow*. New York: Farrar, Straus and Giroux.

Kirby, S. (2001) 'Spontaneous Evolution of Linguistic Structure—an Iterated Learning Model of the Emergence of Regularity and Irregularity', IEEE Transactions *on Evolutionary Computation*, 5: 102–10.

—— et al. (2008) 'Cumulative Cultural Evolution in the Laboratory: An Experimental Approach to the Origins of Structure in Human Language', *Proceedings of the National Academy of Sciences United States of America*, 105: 10681–6.

—— (2014) 'Iterated Learning and the Evolution of Language', *Current Opinion in Neurobiology*, 28: 108–14.

Meulman, N. et al. (2015) 'Age effects in L2 grammar processing as revealed by ERPs and how (not) to study them', *PLoS One*, 10/12: e0143328.

Mirman, D. (2014) *Growth Curve Analysis and Visualization Using R*. Boca Raton: CRC Press.

Mirman, D. et al. (2008) 'Statistical and Computational Models of the Visual World Paradigm: Growth Curves and Individual Differences' *Journal of Memory and Language*, 59/4: 475–94.

—— (2011) 'Theories of Spoken Word Recognition Deficits in Aphasia: Evidence from Eye-Tracking and Computational Modeling', *Brain and Language*, 117/2: 53–68.

—— (2012) 'Treating Participants (or items) as Random vs. Fixed Effects', *Language & Cognitive Dynamics Laboratory Technical Report* 2012.03.

O'Hara, R. B., and Kotze, D. J. (2010) Do not log-transform count data. *Methods in Ecology and Evolution*, 1/2: 118–22.

Perlman, M. et al. (2015) 'Iconicity can Ground the Creation of Vocal Symbols', *Royal Society Open Science*, 2/8: 150–2.

Pinheiro, J. C., and Bates, D.M. (2000) *Mixed-Effects Models in S and SPLUS*. New York: Springer.

Schielzeth, H. (2010) 'Simple Means to Improve the Interpretability of Regression Coefficients', *Methods in Ecology and Evolution*, 1/2: 103–13.

Schielzeth, H., and Forstmeier, W. (2009) 'Conclusions Beyond Support: Overconfident Estimates in Mixed Models', *Behavioral Ecology*, 20/2: 416–20.

Scott-Phillips, T. C., and Kirby, S. (2010) 'Language Evolution in the Laboratory', *Trends in Cognitive Sciences*, 14: 411–7.

van Rij, J., Hollebrandse, B., and Hendriks, P. (in press). 'Children's Eye Gaze Reveals their Use of Discourse Context in Object Pronoun Resolution'. In: A. Holler, C. Glauw and K. Suckow (eds.) *Empirical Perspectives on Anaphora Resolution*. Berlin: Mouton de Gruyter.

—— (2015) *itsadug: Interpreting Time Series and Autocorrelated Data using GAMMs*. R package version 1.0.1.

Wieling, M. et al. (2011) 'Quantitative Social Dialectology: Explaining Linguistic Variation Geographically and Socially', *PLoS One*, 6/9: e23613.

—— (2014) 'Lexical Differences between Tuscan Dialects and Standard Italian: Accounting for Geographic and Socio-demographic Variation using Generalized Additive Mixed Modeling', *Language*, 90/3: 669–92.

—— (2015) 'Investigating Dialectal Differences using Articulography', *Proceedings of ICPhS* 2015, Glasgow.

Winter, B. (2013) 'Linear Models and Linear Mixed Effects Models in R with Linguistic Applications', *arXiv*: 1308.5499.

Wood, S. (2003) 'Thin Plate Regression Splines', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 65/1: 95–114.

—— (2006) *Generalized Additive Models: An Introduction with R*. Boca Raton: CRC Press.

Zuur, A. F. et al. (2009). *Mixed Effects Models and Extensions in Ecology with R*. New York: Springer.