Bayesian Learning of Latent Representations of Language Structures

Yugo Murawaki
Kyoto University
Graduate School of Informatics
murawaki@i.kyoto-u.ac.jp

We borrow the concept of representation learning from deep learning research, and we argue that the quest for Greenbergian implicational universals can be reformulated as the learning of good latent representations of languages, or sequences of surface typological features. By projecting languages into latent representations and performing inference in the latent space, we can handle complex dependencies among features in an implicit manner. The most challenging problem in turning the idea into a concrete computational model is the alarmingly large number of missing values in existing typological databases. To address this problem, we keep the number of model parameters relatively small to avoid overfitting, adopt the Bayesian learning framework for its robustness, and exploit phylogenetically and/or spatially related languages as additional clues. Experiments show that the proposed model recovers missing values more accurately than others and that some latent variables exhibit phylogenetic and spatial signals comparable to those of surface features.

1. Introduction

1.1 Representation Learning for Linguistic Typology

Beginning with the pioneering research by Greenberg (1963), linguists have taken quantitative approaches to linguistic typology. To propose a dozen cross-linguistic generalizations, called linguistic universals (an example is that languages with dominant VSO [verb–subject–object] order are always prepositional), Greenberg investigated a sample of 30 languages from around the world to correct for phylogenetic and areal effects. Linguistic universals, including those formulated in absolute terms by Greenberg, are rarely exceptionless (Dryer 1998), and therefore they are called statistical universals, as opposed to absolute universals.

While a great amount of effort has been invested into theory construction and careful analysis of field data, typologists have relied on elementary statistical concepts, such as frequency, mode, and deviation from expectation (Nichols 1992; Cysouw 2003). The limitations of superficial statistical analysis become evident especially when one seeks *diachronic* explanations of cross-linguistic variation. Although Greenberg (1978) proposed probabilistic models of language change over time, powerful statistical tools for making inferences were not available. To do so, we need a model with predictive

Submission received: 15 July 2018; revised version received: 28 December 2018; accepted for publication: 8 February 2019.

doi:10.1162/COLI_a_00346

power or the ability to draw from the present distribution generalizations that are applicable to the past. To infer the states of languages in the past, or complex latent structures in general, we need a robust statistical framework, powerful inference algorithms, and large computational resources. A statistical package that meets all of these requirements was not known to typologists in Greenberg's day.

Today, the research community of computational linguistics knows a solution: Bayesian models armed with computationally intensive Markov chain Monte Carlo inference algorithms. Indeed, Bayesian models have been used extensively to uncover complex latent structures behind natural language text in the last decade or so (Goldwater and Griffiths 2007; Griffiths, Steyvers, and Tenenbaum 2007; Goldwater, Griffiths, and Johnson 2009).

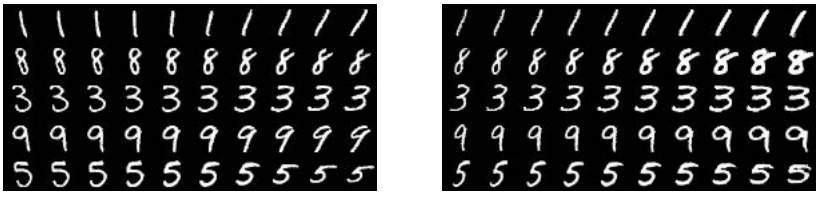
Given this, it is somewhat surprising that with some notable exceptions (Daumé III and Campbell 2007; Daumé III 2009), the application of Bayesian statistics to typological problems has been done largely outside of the computational linguistics community, although computational linguists have recognized the usefulness of typological information for multilingual text processing (Bender 2016; O'Horan et al. 2016). In fact, it is evolutionary biology that has offered solutions to typological questions (Dediu 2010; Greenhill et al. 2010; Dunn et al. 2011; Maurits and Griffiths 2014; Greenhill et al. 2017).

In this article, we demonstrate that **representation learning**, a concept that computational linguists have become familiar with over the past decade, is useful for the study of linguistics typology (Bengio, Courville, and Vincent 2013). Although it has been applied to genomics (Asgari and Mofrad 2015; Tan et al. 2016), to our knowledge, representation learning has not been used in the context of evolution or applied to language data.

The goal of representation learning is to learn useful latent representations of the data (Bengio, Courville, and Vincent 2013). We assume that latent representations exist behind surface representations, and we seek to let a model connect the two types of representations.

To provide intuition, we consider handwritten digits represented by grayscale 28×28 images (LeCun et al. 1998). Each pixel takes one of 256 values. This means that there are $256^{28\times28}$ possible images. However, only a tiny portion of them look like *natural* digits. Such data points must be smoothly connected because natural digits usually continue to look natural even if small modifications are added to them (for example, slightly rotating the images). These observations lead us to the manifold hypothesis: The data reside on low-dimensional manifolds embedded in a high-dimensional space, and thus must be able to be represented by a relatively small number of latent variables. Moreover, good latent representations must disentangle underlying abstract factors, as illustrated in Figure 1. One latent variable represents, say, the angle, while another one smoothly controls the width of digits (Chen et al. 2016). As these factors demonstrate, modification of one latent variable affects multiple pixels at once.

Our key idea is that the same argument applies to typological data, although typological features are much more informative than the pixels in an image. Combining typological features together, we can map a given language to a point in high-dimensional space. What Greenbergian universals indicate is that *natural* languages are not evenly distributed in the space. Most Greenbergian universals are implicational, that is, given in the form of "if x holds, then y also holds." In other words, the combination of $(x, \neg y)$ is non-existent (absolute universals) or rare (statistical universals). In addition, languages have evolved gradually, and many of them are known to share common ancestors. If we accept the uniformitarian hypothesis, that is, the assumption that universals discovered in modern languages should also apply to past languages (Croft 2002),



(a) (b)

Figure 1
Two examples of learned good latent representations for handwritten digits. For each row, one latent variable is gradually changed from left to right whereas the other latent variables are fixed. The latent variables manipulated in (1a) and (1b) appear to control the angle and width of the digits, respectively. The figures are taken from Figure 2 of Chen et al. (2016).

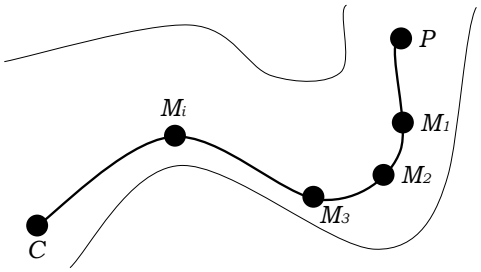


Figure 2
The manifold hypothesis in the context of language evolution. *Natural* languages concentrate around a small subspace whose approximate boundaries are indicated by the two thin lines. Not only the modern language C but also its ancestor P and the intermediate languages $M_1, M_2, \cdots, M_i, \cdots$ must be in the subspace.

a smooth line of *natural* languages must be drawn between a modern language and its ancestor, and by extension, between any pair of phylogenetically related languages, as illustrated in Figure 2. Thus, we expect languages to lie on a smooth lower-dimensional manifold. Just like the angle of a digit, a latent variable must control multiple surface features at once.

1.2 Diachrony-Aware Bayesian Representation Learning

The question, then, is how to turn this idea into a concrete computational model. The concept of representation learning was popularized in the context of deep learning, with deep autoencoders being typical models (Hinton and Salakhutdinov 2006). Accordingly, we previously proposed an autoencoder-based neural network model for typological data (Murawaki 2015). With follow-up experiments, however, we later found that the model suffered from serious problems.

The first problem is overfitting. Neural network methods are known to be data-hungry. They can approximate a wide variety of functions by simply combining general-purpose components (Hornik 1991), but the flexibility is obtained at the cost of requiring very large amounts of data. The database available to us is a matrix

where languages are represented as rows and features as columns (Haspelmath et al. 2005). The number of languages is on the order of 1,000 and the number of features is on the order of 100. This precious database summarizes decades of work by various typologists, but from the viewpoint of machine learning, it is not very large.

More importantly, the typological database is characterized by an alarmingly large number of missing values. Depending on how we perform preprocessing, only 20% to 30% of the items are present in the language–feature matrix. The situation is unlikely to change in the foreseeable future for a couple of reasons. Of the thousands of languages in the world, there is ample documentation for only a handful. Even if grammatical sketches are provided by field linguists, it is not always easy for non-experts to determine the appropriate value for a given feature because typological features are highly theory-dependent. If one manages to provide some missing values, they are also likely to add previously uncovered languages to the database, with few features present. The long tail remains long. Thus there seems to be no way of escaping the problem of missing values.

Here, we combine several methods to cope with this problem. All but one can be collectively referred to as Bayesian learning. We replace the general-purpose neural network with a carefully crafted generative model that has a smaller number of model parameters. We apply prior distributions to the model parameters to penalize extreme values. In inference, we do not rely on a single point estimate of model parameters but draw multiple samples from the posterior distribution to account for uncertainty.[1]

The last part of the proposed method can be derived when we re-interpret implicational universals in terms of the language–feature matrix: They focus on dependencies between columns and thus can be referred to as inter-feature dependencies. We can also exploit dependencies between rows, or inter-language dependencies. It is well known that the values of a typological feature do not distribute randomly in the world but reflect vertical (phylogenetic) transmissions from parents to children and horizontal (spatial or areal) transmissions between populations (Nichols 1992). For example, languages of mainland Southeast Asia, such as Hmong, Thai, and Vietnamese, are known for having similar tonal systems even though they belong to different language families (Enfield 2005). For this reason, combining inter-language dependencies with inter-feature dependencies is a promising solution to the problem of missing values. Whereas inter-feature dependencies are synchronic in nature, inter-language dependencies reflect diachrony, at least in an indirect manner. Thus, we call the combined approach **diachrony-aware learning**.

As a building block, we use a Bayesian autologistic model that takes both the vertical and horizontal factors into consideration (Murawaki and Yamauchi 2018). Just like the familiar logistic model, the autologistic model assumes that a dependent random variable (a language in our case) depends probabilistically on explanatory variables. The difference is that explanatory variables themselves are languages that are to be stochastically explained by other languages. The motivation behind this is that languages that are related either vertically or horizontally must be predictive of the language in question. Thus, languages are dependent on each other and form a

1 The combination of neural networks and Bayesian learning is actively studied (Welling and Teh 2011) and is applied to natural language tasks (Gan et al. 2017). One thing common to these studies, including ours, is the use of Hamiltonian Monte Carlo (HMC) or variants of it for inference. Bayesian neural networks use online extensions to HMC because scalability is a vital concern. In contrast, we use vanilla HMC because our database is relatively small.

neighbor graph in which every pair of interdependent languages is connected. A major advantage of the autologistic model over the standard tree model (Gray and Atkinson 2003; Bouckaert et al. 2012) is the ability to integrate the vertical and horizontal factors into a single model by simply using two neighbor graphs (Towner et al. 2012).

To do so, we make use of two additional resources: (1) a phylogenetic neighbor graph of languages that can be generated by connecting every pair of languages in each language family and (2) a spatial neighbor graph that connects languages within a specified distance.

A problem with the combined approach is that the model for inter-language dependencies, in its original form, cannot be integrated into the model for inter-feature dependencies. They both explain how the surface language–feature matrix is generated, even though only one generative story can exist. To resolve the conflict, we incorporate the autologistic model at the level of latent representations, rather than surface features, with the reasonable assumption that phylogenetically and/or spatially close languages tend to share the same latent variables in addition to the same surface features. In the end, the integrated Bayesian generative model first generates the latent representations of languages using inter-language dependencies, and then generates the surface representations of languages using inter-feature dependencies, as summarized in Figure 3.

Experiments show that the proposed Bayesian model recovers missing values considerably more accurately than other models. In addition, the integrated model consistently outperforms baseline models that exploit only one of the two types of dependencies, demonstrating the complementary nature of inter-feature and inter-language dependencies.

Since autologistic models require variables to be discrete, we inevitably adopt *binary* latent representations. We call our latent variables **linguistic parameters** for their superficial resemblance to parameters in the principles-and-parameters framework of generative grammar (Chomsky and Lasnik 1993) and for other reasons. A side effect of the discreteness constraint is good interpretability of linguistic parameters, in comparison with that of the continuous representations of Murawaki (2015). To demonstrate this, we project linguistic parameters on a world map and show that at least some of them exhibit phylogenetic and spatial signals comparable to those of surface features. Also, because both the surface and latent representations are discrete, linguistic parameters can readily be used as a substitute for surface features and therefore have a wide range of potential applications, including tree-based phylogenetic inference (Gray and Atkinson 2003; Bouckaert et al. 2012; Chang et al. 2015).

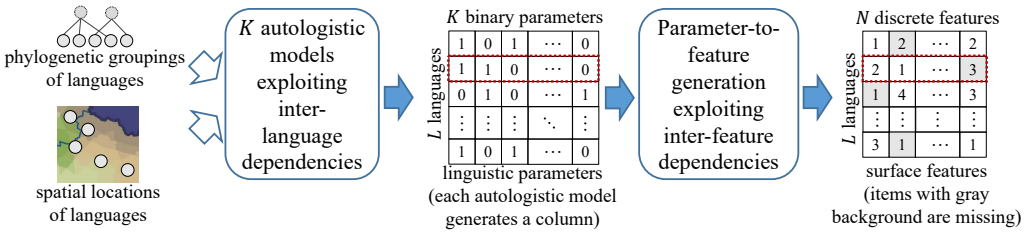


Figure 3
Overview of the proposed Bayesian generative model. Dotted boxes indicate the latent and surface representations of the same language. Solid arrows show the direction of stochastic generation. Symbols used here are explained in Table 1.

2. Background

2.1 Greenbergian Universals

Since Greenberg (1963), various interdependencies among typological features have been observed across the world's languages. For example, if a language takes a verb before an object (VO), then it takes postnominal relative clauses (NRel) (VO → NRel, in shorthand), and a related universal, RelN → OV, also holds (Dryer 2011). Such cross-linguistic generalizations are specifically called **Greenbergian universals**, as opposed to Chomskyan universals, which we discuss in the next section. A Bayesian model for discovering Greenbergian universals was presented by Daumé III and Campbell (2007).

Greenbergian universals indicate that certain combinations of features are unnatural. Moreover, Greenberg (1978) discussed how to extend the synchronic observations to diachronic reasoning: Under the uniformitarian hypothesis, languages must have changed in a way such that they avoid unnatural combinations of features.

Despite these highly influential observations, most computational models of typological data assume independence between features (Daumé III 2009; Dediu 2010; Greenhill et al. 2010; Murawaki 2016; Greenhill et al. 2017). These methods are at risk for reconstructing typologically unnatural languages. A rare exception is Dunn et al. (2011), who extended Greenberg's idea by applying a phylogenetic model of correlated evolution (Pagel and Meade 2006).

Both Greenberg (1963, 1978) and Dunn et al. (2011) focused on *pairs* of features. However, the dependencies between features are not limited to feature pairs (Tsunoda, Ueda, and Itoh 1995; Itoh and Ueda 2004). The order of relative clauses, just mentioned above, has connections to the order of adjective and noun (AdjN or NAdj), in addition to the order of object and verb, as two universals, RelN → AdjN and NAdj → NRel, are known to hold well (Dryer 2011).

Limiting the scope of research to feature pairs is understandable given that a combination of three or more features is often beyond human comprehension. Even for computers, extending a model of feature pairs (Dunn et al. 2011) to multiple features is hampered by computational intractability due to combinatorial explosion. What we propose here is a computationally tractable way to handle multiple inter-feature dependencies. We map interdependent variables to latent variables that are independent from each other by assumption. If we perform inference in the latent space (for example, reconstructing ancestral languages from their descendants) and then project the data back to the original space, we can handle inter-feature dependencies in an implicit manner.

2.2 Chomskyan Universals

Thanks to its tradition of providing concrete and symbolic representations to latent structures of languages, generative grammar has constantly given inspiration to computational linguists. At the same time, however, it is a source of frustration because, with Optimality Theory (Prince and Smolensky 2008) being a notable exception, it rarely explores how disambiguation is performed. Linguistic typology is no exception. Symbolic latent representations behind surface patterns are proposed (Baker 2001), but they have trouble explaining disharmonic languages around the world (Boeckx 2014).

A Chomskyan explanation for typological variation is (macro)parameters, which are part of the **principles and parameters** (P&P) framework (Chomsky and Lasnik 1993). In this framework, the structure of a language is explained by (1) a set of universal

principles that are common to all languages and (2) a set of parameters whose values vary among languages. Here we skip the former because our focus is on structural variability. According to P&P, if we give specific values to all the parameters, then we obtain a specific language. Each parameter is binary and, in general, sets the values of multiple surface features in a deterministic manner. For example, the head directionality parameter is either `head-initial` or `head-final`. If `head-initial` is chosen, then surface features are set to `VO`, `NAdj`, and `Prepositions`; otherwise, the language in question becomes `OV`, `AdjN`, and `Postpositions` (Baker 2001). Baker (2001) discussed a number of parameters, such as the head directionality, polysynthesis, and topic-prominent parameters.

Our design decision to use *binary* latent representations is partly inspired by the parameters of generative grammar. We also borrow the term **parameter** from this research field, due to a conflict in terminology. **Features** usually refer to latent representations in the machine learning community (Griffiths and Ghahramani 2011; Bengio, Courville, and Vincent 2013). Unfortunately, the word *feature* is reserved for surface variables in the present study, and we need another name for latent variables. We admit the term *parameter* is confusing because, in the context of machine learning, it refers to a variable tied to the model itself, rather than its input or output. For clarity, we refer to binary latent representations as **linguistic parameters** throughout this article. A parameter of the model is referred to as a **model parameter**.

It should be noted that we do not intend to present the proposed method as a computational procedure to induce P&P parameters. Although our binary latent representations are partly inspired by P&P, their differences cannot be ignored. There are at least five differences between the P&P framework and the proposed model. First, whereas the primary focus of generative linguists is put on morphosyntactic characteristics of languages, the data sets we used in the experiments are not limited to them.

Second, Baker (2001) presented a hierarchical organization of parameters (see Figure 6.4 of Baker [2001]). However, we assume independence between linguistic parameters. Introducing a hierarchical structure to linguistic parameters is an interesting direction to explore, but we leave it for future work.

Third, whereas P&P hypothesizes deterministic generation, the proposed model *stochastically* generates a language's features from its linguistic parameters. This choice appears to be inevitable because obtaining exceptionless relations from real data is virtually impossible.

Fourth, a P&P parameter typically controls a very small set of features. In contrast, if our linguistic parameter is turned on, it more or less modifies all the feature generation probabilities. However, we can expect a small number of linguistic parameters to dominate the probabilities because weights are drawn from a heavy-tailed distribution, as we describe in Section 4.1.

Lastly, our linguistic parameters are asymmetric in the sense that they do not operate at all if they are off. Although the marked–unmarked relation comes about as a natural consequence of incorporating binary variables into a computational model, this is not necessarily the case with P&P. For example, the head directionality parameter has two values, `head-initial` and `head-final`, and it is not clear which one is marked and which one is unmarked. In the Bayesian analysis of mixture models, mixture components are known to be unidentifiable because the posterior distribution is invariant to permutations in the labels (Jasra, Holmes, and Stephens 2005). In our model, the on and off of a linguistic parameter is not exactly swappable, but it is likely that a point in the search space where `head-initial` is treated as the marked form is separated by deep valleys from another point where `head-final` is treated as the marked form. If this is

the case, a Gibbs sampler generally cannot cross the valleys. We discuss this point again in Section 7.3.

2.3 Functional Explanations

Needless to say, Greenbergian typologists themselves have provided explanations for cross-linguistic patterns although, unlike generative linguists, they generally avoid using *metaphysical* representations. Such explanations can be collectively referred to as functional explanations (Haspelmath 2008b).

One major type of functional explanation is synchronic in nature and often is a matter of economy. For example, several languages exhibit an adnominal alienability split, that is, the use of different possessive constructions for **inalienable** nouns (e.g., my arm) and **alienable** nouns (e.g., your car). An implicational universal for the split is:

> If a language has an adnominal alienability split, and one of the constructions is overtly coded while the other one is zero-coded, it is always the inalienable construction that is zero-coded, while the alienable construction is overtly coded. (Haspelmath 2008a)

Haspelmath (2008a) points to the fact that inalienable nouns occur as possessed nouns much more frequently than alienable nouns. This means that inalienable nouns are more predictable and, consequently, a shorter (even zero) marker is favored for efficiency.

Another type is diachronic explanations. According to this view, at least some patterns observed in surface features arise from common paths of diachronic development (Anderson 2016). An important factor of diachronic development is grammaticalization, by which content words change into function words (Heine and Kuteva 2007). For example, the correlation between the order of adposition and noun and the order of genitive and noun might be explained by the fact that adpositions are often derived from nouns.

Regardless of whether they are synchronic or diachronic, functional explanations imply that unattested languages may simply be improbable but not impossible (Haspelmath 2008b). Because of the stochastic nature of the proposed model, we are more closely aligned with functionalists than with generative linguists. Note that the proposed model only describes patterns found in the data. It does not explain the underlying cause-and-effect mechanisms, although we hope that it can help linguists explore them.

2.4 Vertical and Horizontal Transmission

The standard model for phylogenetic inference is the tree model, where a trait is passed on from parent to child with occasional modifications. In fact, the recent success in the applications of statistical models to historical linguistic problems is largely attributed to the tree model (Gray and Atkinson 2003; Bouckaert et al. 2012), although the applications are subject to frequent criticism (Chang et al. 2015; Pereltsvaig and Lewis 2015). In linguistic typology, however, a non-tree-like mode of evolution has emerged as one of the central topics (Trubetzkoy 1928; Campbell 2006). Typological features, like loanwords, can be borrowed by one language from another, and as a result, vertical (phylogenetic) signals are obscured by horizontal (spatial) transmission.

The task of incorporating both vertical and horizontal transmissions within a statistical model of language evolution is notoriously challenging because of the excessive flexibility of horizontal transmissions. This is the reason why previously proposed models are coupled with some very strong assumptions—for example, that a reference

tree is given a priori (Nelson-Sathi et al. 2011) and that horizontal transmissions can be modeled through time-invariant areal clusters (Daumé III 2009).

Consequently, we pursue a line of research in linguistic typology that draws on information on the current distribution of typological features without explicitly requiring the reconstruction of previous states (Nichols 1992, 1995; Parkvall 2008; Wichmann and Holman 2009). The basic assumption is that if the feature in question is vertically stable, then a phylogenetically defined group of languages will tend to share the same value. Similarly, if the feature in question is horizontally diffusible, then spatially close languages would be expected to frequently share the same feature value. Because the current distribution of typological features is more or less affected by these factors, the model needs to take both vertical and horizontal factors into account.

Murawaki and Yamauchi (2018) adopted a variant of the autologistic model, which had been widely used to model the spatial distribution of a feature (Besag 1974; Towner et al. 2012). The model was also used to impute missing values because the phylogenetic and spatial neighbors of a language had some predictive power over its feature values. Our assumption in this study is that the same predictive power applies to latent representations.

3. Data and Preprocessing

3.1 Input Specifications

The proposed model requires three types of data as the input: (1) a language–feature matrix, (2) a phylogenetic neighbor graph, and (3) a spatial neighbor graph. Table 1 lists the major symbols used in this article.

Let L and N be the numbers of languages and surface features, respectively. The language–feature matrix $X \in \mathbb{N}^{L \times N}$ contains discrete items. A substantial portion of the items may be missing. $x_{l,n}$ denotes the value of feature n for language l. Features can be classified into three types: (1) binary ($x_{l,n} \in \{0, 1\}$), (2) categorical ($x_{l,n} \in \{1, 2, \cdots, F_n\}$, where F_n is the number of distinct values), and (3) count ($x_{l,n} \in \{0, 1, 2, \cdots\}$).

A neighbor graph is an undirected graph in which each node represents a language. The graph connects every pair of languages that are related in some way and thus are likely to be similar to some degree. A phylogenetic neighbor graph connects phyloge-

Table 1
Notations. Corresponding item indices are in parentheses.

L	(l)	# of languages
K	(k)	# of linguistic parameters (given a priori)
M	(m)	# of model parameters of W, $\tilde{\Theta}$, and Θ
N	(n)	# of surface discrete linguistic features
F_n		# of distinct values for categorical feature n
$A = \{(v_k, h_k, u_k) \mid k \in \{1, \cdots, K\}\}$		Model parameters for the autologistic models
$Z \in \{0, 1\}^{L \times K}$		Binary latent parameter matrix
$W \in \mathbb{R}^{K \times M}$		Weight matrix
$\tilde{\Theta} \in \mathbb{R}^{L \times M}$		Unnormalized model parameter matrix
$\Theta \in (0, 1)^{L \times M}$		Normalized model parameter matrix
$X \in \mathbb{N}^{L \times N}$		Surface discrete linguistic feature matrix

netically related languages, while a spatial neighbor graph connects spatially close pairs of languages.

3.2 Preprocessing

Although any database that meets the requirements described in Section 3.1 can be used, we specifically tested two typological databases in the present study: (1) the online edition[2] of the *World Atlas of Language Structures* (WALS) (Haspelmath et al. 2005) and (2) Autotyp 0.1.0 (Bickel et al. 2017).

WALS (Haspelmath et al. 2005) is a large database of typological features compiled by dozens of typologists. Since its online version was released in 2008, it has been occasionally used by the computational linguistics community (Naseem, Barzilay, and Globerson 2012; O'Horan et al. 2016; Bender 2016). WALS covers a wide range of linguistic domains (called "areas" in WALS), such as phonology, morphology, nominal categories, and word order. All features are categorically coded. For example, Feature 81A, "Order of Subject, Object and Verb" has seven possible values: `SOV`, `SVO`, `VSO`, `VOS`, `OVS`, `OSV`, and `No dominant order`, and each language is assigned one of these seven values.

We downloaded a CSV file that contained metadata and feature values for each language. Sign languages were dropped because they were too different from spoken languages. Pidgins and creoles were also removed from the matrix because they belonged to the dummy language family "other." We imputed some missing values that could trivially be inferred from other features. Feature 144D, "The Position of Negative Morphemes in SVO Languages" is an example. Because it is only applicable to SVO languages, languages for which the value of Feature 81A is `SOV` are given the special value `Undefined`. We then removed features that covered fewer than 150 languages. We manually classified features into binary and categorical features (no count features were present in WALS) and replaced text-format feature values with numerical ones.

WALS provides two-level phylogenetic groupings: **family** (upper) and **genus** (lower). For example, English belongs to the Indo-European family and to its subgroup (genus), Germanic. Genera are designed to be roughly comparable taxonomic groups so that they facilitate cross-linguistic comparison (Dryer 1989). Following Murawaki and Yamauchi (2018), we constructed a phylogenetic neighbor graph by connecting every pair of languages within each genus.

WALS associates each language with single-point geographical coordinates (longitude and latitude). Following Murawaki and Yamauchi (2018), we constructed a spatial neighbor graph by linking all language pairs that were located within a distance of $R = 1,000$ km.

Autotyp (Bickel et al. 2017) is a smaller database, and it appears to be more coherent because a smaller number of typologists led its construction. It is a mixture of raw data and automatically aggregated data and covers finer-grained domains (called "modules" in Autotyp) such as alignment per language, locus per language, and valence per language. In Autotyp, domains are classified into three types: (1) single entry per language, (2) single aggregated entry per language, and (3) multiple entries per language. We only used the first two types of domains, in which a language is given a single value per feature. As the name suggests, features belonging to the last type of domains have

2 `http://wals.info/`.

Table 2
Data set specifications after preprocessing.

		WALS	Autotyp
L	(# of languages)	2,607	1,063
K	(# of linguistic parameters)	50 or 100	
M	(# of model parameters of W, $\tilde{\Theta}$, and Θ)	760	958
N	(# of discrete features)	152	372
# of binary features		14	229
# of categorical features		138	118
# of count features		0	25
Proportion of items present in the language–feature matrix (%)		19.98	21.54
# of phylogenetic neighbors on average		30.77	7.43
# of spatial neighbors on average		89.10	38.53

multiple values in general, and the number of distinct combined values can reach the order of 100.

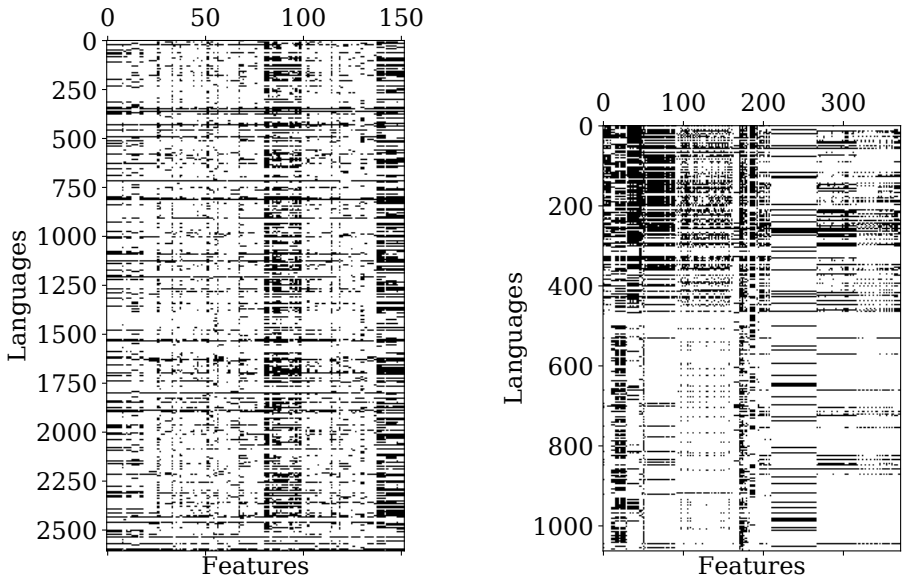
We downloaded the data set from the GitHub repository.[3] In addition to dropping sign languages, and creole and mixed languages, which were all marked as such in the metadata, we manually removed ancient languages. Languages without phylogenetic information or geographical coordinate points were also removed. We manually classified features into binary, categorical, and count features[4] and assigned numerical codes to the values of the binary and categorical features. We then removed features that covered fewer than 50 languages.

The phylogenetic and spatial neighbor graphs for Autotyp data were constructed as were done for WALS. One difference was that Autotyp did not have a single phylogenetic level comparable to WALS's genera. It instead provides six non-mandatory metadata fields: majorbranch, stock, subbranch, subsubbranch, lowestsubbranch, and quasistock. We attempted to create genus-like groups by combining these fields, but the results were far from perfect and are subject to future changes.

Table 2 summarizes the results of preprocessing (K and M are introduced later). We can see that although we removed low-coverage features, only about 20% of items were present in the language–feature matrices X. Figure 4 visualizes X. It is evident that we were dealing with a type of missing values called **missing not at random**. Data gaps are not random because both languages and features exhibit power-law behavior, that is, a small number of high-coverage languages (features) are contrasted with heavy tails of low-coverage languages (features). What is worse, the lack of one feature is predictive of the lack of some others because typologists have coded multiple related features at once.

3 `https://github.com/autotyp/autotyp-data`.

4 We discarded ratio features whose values ranged from 0 to 1, inclusive. Although it is possible to model ratio data, it is generally not a good idea because the "raw" features from which the ratio is calculated are more informative.



(a) WALS. (b) Autotyp.

Figure 4
Missing values in the language–feature matrices. Items present are filled black.

4. Bayesian Generative Model

Our goal is to induce a binary latent parameter matrix $Z \in \{0,1\}^{L \times K}$ from the observed portion of the language–feature matrix X. K is the number of linguistic parameters and is to be specified a priori. To obtain Z, we first define a probabilistic generative model that describes how Z is stochastically generated and how X is generated from Z. After that, we devise a method to infer Z, as we explain in Section 5. For now, we do not need to care about missing values.

As shown in Figure 3, the model assumes a two-step generation process: It exploits inter-language dependencies for the first part and inter-feature dependencies for the second part. Accordingly, the joint distribution is given as

$$P(A, Z, W, X) = P(A)P(Z|A)P(W)P(X|Z, W) \tag{1}$$

where hyperparameters are omitted for brevity. A is a set of model parameters that control the generation of Z, whereas W is a weight matrix that connects Z and X.

For ease of description, we trace the generative story *backward* from X. Section 4.1 describes the second part, which is followed by Section 4.2 for the first part.

4.1 Inter-Feature Dependencies

In this section, we describe how the surface feature representations are generated from the binary latent representations. Figure 5 illustrates the process. We use matrix factorization (Srebro, Rennie, and Jaakkola 2005; Griffiths and Ghahramani 2011) to capture inter-feature dependencies. Because the discrete feature matrix X cannot directly be

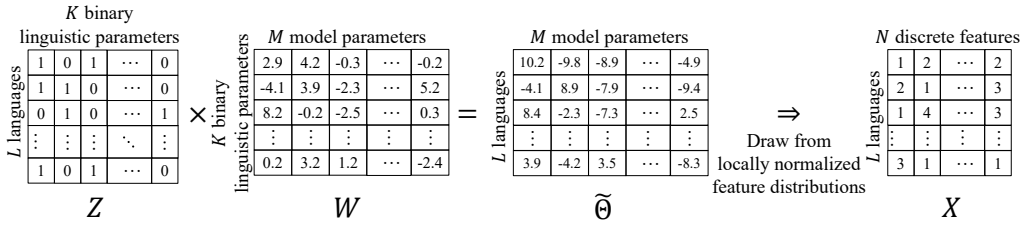


Figure 5
Stochastic parameter-to-feature generation. $\tilde{\Theta} = ZW$ encodes inter-feature dependencies.

decomposed into two matrices, we instead decompose a closely related, unnormalized model parameter matrix $\tilde{\Theta}$. It is $\tilde{\Theta}$ that directly controls the stochastic generation of X.

Recall that $x_{l,n}$ can take a binary, categorical, or count value. As usual, we assume that a binary feature is drawn from a Bernoulli distribution:

$$x_{l,n} \sim \text{Bernoulli}(\theta_{l,f(n,1)}) \tag{2}$$

where $\theta_{l,f(n,1)} \in (0,1)$ is the corresponding model parameter. Because, as we discuss subsequently, one feature can correspond to more than one linguistic parameter, feature n is mapped to the corresponding model parameter index by the function $f(n,i) \in \{1,\cdots,m,\cdots,M\}$. A binary or count feature has one model parameter whereas a categorical feature with F_n distinct values has F_n model parameters. M is the total number of these model parameters. $\theta_{l,m}$ is an item of the normalized model parameter matrix $\Theta \in (0,1)^{L \times M}$.

A categorical value is generated from a categorical distribution:

$$x_{l,n} \sim \text{Categorical}(\theta_{l,f(n,1)},\cdots,\theta_{l,f(n,F_n)}) \tag{3}$$

where $\theta_{l,f(n,i)} \in (0,1)$ and $\sum_{i=1}^{F_n} \theta_{l,f(n,i)} = 1$. As you can see, the categorical feature n has F_n model parameters.

A count value is drawn from a Poisson distribution:

$$x_{l,n} \sim \text{Poisson}(\theta_{l,f(n,1)}) \tag{4}$$

where $\theta_{l,f(n,1)} > 0$. This distribution has mean and variance $\theta_{l,f(n,1)}$.[5]

Θ is obtained by normalizing $\tilde{\Theta} \in \mathbb{R}^{L \times M}$. For binary features, we use the sigmoid function:

$$\theta_{l,f(n,1)} = \text{sigmoid}(\tilde{\theta}_{l,f(n,1)}) = \frac{1}{1+\exp(-\tilde{\theta}_{l,f(n,1)})} \tag{5}$$

5 Alternatively, we can use a negative binomial distribution because it may provide a closer fit by decoupling mean and variance. Another option is to use a Poisson hurdle distribution (Mullahy 1986), which deals with the high occurrence of zeroes in the data.

Similarly, the softmax function is used for categorical features:

$$\theta_{l,f(n,i)} = \text{softmax}_i(\tilde{\theta}_{l,f(n,1)}, \cdots, \tilde{\theta}_{l,f(n,F_n)}) = \frac{\exp(\tilde{\theta}_{l,f(n,i)})}{\sum_{i'=1}^{F_n} \exp(\tilde{\theta}_{l,f(n,i')})} \tag{6}$$

and the softplus function for count features:

$$\theta_{l,f(n,1)} = \text{softplus}(\tilde{\theta}_{l,f(n,1)}) = \log(1 + \exp(\tilde{\theta}_{l,f(n,1)})) \tag{7}$$

The unnormalized model parameter matrix $\tilde{\Theta}$ is a product of the binary latent parameter matrix Z and the weight matrix W. The generation of Z is described in Section 4.2. Each item of $\tilde{\Theta}$, $\tilde{\theta}_{l,m}$, is language l's m-th unnormalized model parameter. It is affected only by linguistic parameters with $z_{l,k} = 1$ because

$$\tilde{\theta}_{l,m} = \sum_{k=1}^{K} z_{l,k} w_{k,m} \tag{8}$$

To investigate how categorical features are related to each other, we combine Equations (8) and (6). We obtain

$$\begin{aligned} \theta_{l,f(n,i)} &\propto \exp\left(\sum_{k=1}^{K} z_{l,k} w_{k,f(n,i)}\right) \\ &= \prod_{k=1}^{K} \exp(z_{l,k} w_{k,f(n,i)}) \end{aligned} \tag{9}$$

We can see from Equation (9) that this is a product-of-experts model (Hinton 2002). If $z_{l,k} = 0$, the linguistic parameter k has no effect on $\theta_{l,f(n,i)}$ because $\exp(z_{l,k} w_{k,f(n,i)}) = 1$. Otherwise, if $w_{k,f(n,i)} > 0$, it makes $\theta_{l,f(n,i)}$ larger, and if $w_{k,f(n,i)} < 0$, it lowers $\theta_{l,f(n,i)}$. Suppose that for the linguistic parameter k, a certain group of languages takes $z_{l,k} = 1$. If two categorical feature values (n_1, i_1) and (n_2, i_2) have large positive weights ($w_{k,f(n_1,i_1)} > 0$ and $w_{k,f(n_2,i_2)} > 0$), the pair must often co-occur in these languages. Likewise, the fact that two feature values do not co-occur can be encoded as a positive weight for one value and a negative weight for the other.

Binary and count features are more straightforward because both the sigmoid and softplus functions take a single argument and increase monotonically. For a binary feature, if $z_{l,k} = 1$ and $w_{k,f(n,i)} > 0$, then $\theta_{l,f(n,i)}$ approaches 1. $w_{k,f(n,i)} < 0$ makes $\theta_{l,f(n,i)}$ closer to 0. The best fit for the count data is obtained when the value equals the mode of the Poisson distribution, which is close to $\theta_{l,f(n,i)}$.

Each item of W, $w_{k,m}$, is generated from a Student t-distribution with 1 degree of freedom. We choose this distribution for two reasons. First, it has heavier tails than the Gaussian distribution and allows some weights to fall far from 0. Second, our inference algorithm demands that the negative logarithm of the probability density function

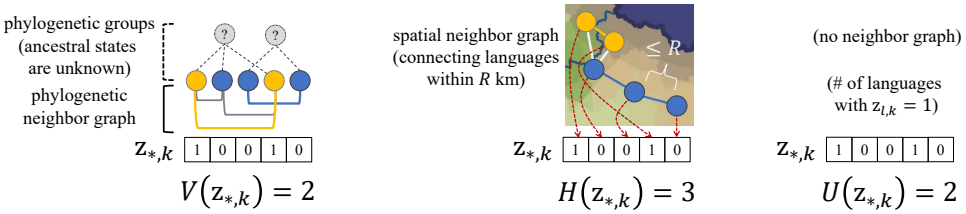


Figure 6
Neighbor graphs and counting functions used to encode inter-language dependencies.

be differentiable, as explained in Section 5.2. The t-distribution satisfies the condition whereas the Laplace distribution does not.[6]

4.2 Inter-Language Dependencies

The autologistic model (Murawaki and Yamauchi 2018) for the linguistic parameter k generates a column of Z, $\mathbf{z}_{*,k} = (z_{1,k}, \cdots, z_{L,k})$. To construct the model, we use two neighbor graphs and the corresponding three counting functions, as illustrated in Figure 6. $V(\mathbf{z}_{*,k})$ returns the number of pairs sharing the same value in the phylogenetic neighbor graph, and $H(\mathbf{z}_{*,k})$ is the spatial equivalent of $V(\mathbf{z}_{*,k})$. $U(\mathbf{z}_{*,k})$ gives the number of languages that take the value 1.

We now introduce the following variables: vertical stability $v_k > 0$, horizontal diffusibility $h_k > 0$, and universality $u_k \in (-\infty, \infty)$ for each linguistic parameter k. The probability of $\mathbf{z}_{*,k}$ conditioned on v_k, h_k, and u_k is given as

$$P(\mathbf{z}_{*,k} \mid v_k, h_k, u_k) = \frac{\exp\left(v_k V(\mathbf{z}_{*,k}) + h_k H(\mathbf{z}_{*,k}) + u_k U(\mathbf{z}_{*,k})\right)}{\sum_{\mathbf{z}'_{*,k}} \exp\left(v_k V(\mathbf{z}'_{*,k}) + h_k H(\mathbf{z}'_{*,k}) + u_k U(\mathbf{z}'_{*,k})\right)} \tag{10}$$

The denominator is a normalization term, ensuring that the sum of the distribution equals one.

The autologistic model can be interpreted in terms of the competition associated with the 2^L possible assignments of $\mathbf{z}_{*,k}$ for the probability mass 1. If a given value, $\mathbf{z}_{*,k}$, has a relatively large $V(\mathbf{z}_{*,k})$, then setting a large value for v_k enables it to appropriate fractions of the mass from its weaker rivals. However, if too large a value is set for v_k, then it will be overwhelmed by its stronger rivals.

To acquire further insights into the model, let us consider the probability of language l taking value $b \in \{0, 1\}$, conditioned on the rest of the languages, $\mathbf{z}_{-l,k} = (z_{1,k}, \cdots, z_{l-1,k}, z_{l+1,k}, \cdots, z_{L,k})$:

$$P(z_{l,k} = b \mid \mathbf{z}_{-l,k}, v_k, h_k, u_k) \propto \exp\left(v_k V_{l,k,b} + h_k H_{l,k,b} + u_k b\right) \tag{11}$$

6 Alternatively, we can explicitly impose sparsity by generating another binary matrix Z_W and replacing W with $Z_W \odot W$, where $\odot$ denotes element-wise multiplication.

where $V_{l,k,b}$ is the number of language l's phylogenetic neighbors that assume the value b, and $H_{l,k,b}$ is its spatial counterpart. $P(z_{l,k} = b \mid \mathrm{z}_{-l,k}, v_k, h_k, u_k)$ is expressed by the weighted linear combination of the three factors in the log-space. It will increase with a rise in the number of phylogenetic neighbors that assume the value b. However, this probability depends not only on the phylogenetic neighbors of language l, but it also depends on its spatial neighbors and on universality. How strongly these factors affect the stochastic selection is controlled by v_k, h_k, and u_k.

Recall that matrix Z has K columns. Accordingly, we have K autologistic models:

$$P(Z \mid A) = \prod_{k=1}^{K} P(\mathrm{z}_{*,k} \mid v_k, h_k, u_k) \tag{12}$$

The model parameter set A can be decomposed in a similar manner:

$$P(A) = \prod_{k=1}^{K} P(v_k)P(h_k)P(u_k) \tag{13}$$

Their prior distributions are: $v_k \sim \mathrm{Gamma}(\kappa, \theta)$, $h_k \sim \mathrm{Gamma}(\kappa, \theta)$, and $u_k \sim \mathcal{N}(0, \sigma^2)$. They complete the generative story. In the experiments, we set shape $\kappa = 1$, scale $\theta = 1$, and standard deviation $\sigma = 10$. These priors are not non-informative, but they are sufficiently gentle in the regions where these model parameters typically reside.

An extension of the model is to set $\mathrm{z}_{*,K} = (1, \cdots, 1)$ for the last linguistic parameter K. Consequently, the autologistic model is dropped from the linguistic parameter K (while $K - 1$ autologistic models remain). With this modification, the weight vector $\mathrm{w}_{K,*} = (w_{K,1}, \cdots, w_{K,M})$ is activated for all languages and serves as a bias term. We used this version of the model in the experiments.

Finally, let us consider a simplified version of the model. If we set $v_k = h_k = 0$, Equation (11) is reduced to

$$P(z_{l,k} = b \mid \mathrm{z}_{-l,k}, v_k, h_k, u_k) = P(z_{l,k} = b \mid u_k) = \frac{\exp(u_k b)}{1 + \exp(u_k)} \tag{14}$$

We can see that the generation of $z_{l,k}$ no longer depends on $\mathrm{z}_{-l,k}$ and is a simple Bernoulli trial with probability $\exp(u_k)/(1 + \exp(u_k))$.[7]

7 Indian buffet processes (Griffiths and Ghahramani 2011) are the natural choice for modeling binary latent matrices (Görür, Jäkel, and Rasmussen 2006; Knowles and Ghahramani 2007; Meeds et al. 2007; Doyle, Bicknell, and Levy 2014). The Indian buffet process (IBP) is appealing for its ability to adjust the number of linguistic parameters to data. A linguistic parameter k is called an **active** parameter if it has one or more languages with $z_{l,k} = 1$. The key property of the IBP is that although there are an unbounded number of linguistic parameters, the number of active linguistic parameters K^+ is finite. K^+ changes during posterior inference. It is decremented when one linguistic parameter becomes inactive. Similarly, it is incremented when $z_{l,k}$ changes from 0 to 1 for an inactive linguistic parameter k.

We modeled $P(Z \mid A)$ using an IBP in a preliminary study but later switched to the present model for two reasons. First, it is difficult to extend an IBP to incorporate inter-language dependencies. It appears that we have no choice but to replace the IBP with the product of the autologistic models. Second, the nonparametric model's adaptability did not work in our case. In theory, Gibbs sampling converges to a stationary distribution after a sufficiently large number of iterations. However, we observed that the number of active linguistic parameters heavily depended on its initial value K_0 because it was very rare for additional linguistic parameters to survive as active linguistic parameters. For this reason, the number of linguistic parameters for the present model, K, is given a priori and is fixed throughout posterior inference.

5. Posterior Inference

Once the generative model is defined, we want to infer the binary latent matrix Z together with other latent variables. With a slight abuse of notation, let X be disjointly decomposed into the observed portion X^{obs} and the remaining missing portion X^{mis}. Formally, the posterior probability is given as

$$P(A, Z, W, X^{\text{mis}} \mid X^{\text{obs}}) \propto P(A, Z, W, X^{\text{mis}} \cup X^{\text{obs}}) \tag{15}$$

As usual, we use Gibbs sampling to draw samples from the posterior distribution. Given observed values $x_{l,n}$, we iteratively update $z_{l,k}$, v_k, h_k, u_k, and $\mathrm{w}_{k,*}$ as well as missing values $x_{l,n}$.

Update missing $x_{l,n}$. $x_{l,n}$ is sampled from Equations (2), (3), or (4), depending on the type of feature n.

Update $z_{l,k}$ and $\mathrm{x}^{\text{mis}}_{l,*}$. We use the Metropolis-Hastings algorithm to update $z_{l,k}$ and $\mathrm{x}^{\text{mis}}_{l,*}$, the missing portion of $\mathrm{x}_{l,*} = (x_{l,1}, \cdots, x_{l,N})$. We find that updating $\mathrm{x}^{\text{mis}}_{l,*}$ drastically improves the mobility of $z_{l,k}$. The proposal distribution first toggles the current $z_{l,k}$ to obtain the proposal $z'_{l,k}$ (1 if $z_{l,k} = 0$; 0 otherwise). As the corresponding $\mathrm{w}_{k,*} = (w_{k,1}, \cdots, w_{k,M})$ gets activated or inactivated, $\theta'_{l,*} = (\theta'_{l,1}, \cdots, \theta'_{l,M})$ is also updated accordingly. We perform a Gibbs sampling scan on $\mathrm{x}^{\text{mis}}_{l,*}$: Every missing $x_{l,n}$ is sampled from the corresponding distribution with the proposal model parameter(s). The proposal is accepted with probability

$$\min\left(1, \frac{P(z'_{l,k}, \mathrm{x}'_{l,*} \mid -)}{P(z_{l,k}, \mathrm{x}_{l,*} \mid -)} \frac{Q(z_{l,k}, \mathrm{x}^{\text{mis}}_{l,*} \mid z'_{l,k}, \mathrm{x}^{\text{mis}\prime}_{l,*})}{Q(z'_{l,k}, \mathrm{x}^{\text{mis}\prime}_{l,*} \mid z_{l,k}, \mathrm{x}^{\text{mis}}_{l,*})}\right) \tag{16}$$

where conditional parts are omitted for brevity. $P(z_{l,k}, \mathrm{x}_{l,*} \mid -)$ is the probability of generating the current state $(z_{l,k}, \mathrm{x}_{l,*})$, while $P(z'_{l,k}, \mathrm{x}'_{l,*} \mid -)$ is the probability of generating the proposed state $(z'_{l,k}, \mathrm{x}'_{l,*})$, in which $\mathrm{x}^{\text{mis}}_{l,*}$ is updated by the proposal distribution. Q is the proposal function constructed as explained above. Equation (16) can be calculated by combining Equations (11), (2), (3), and (4).

Update v_k, h_k, and u_k. We want to sample v_k (and h_k and u_k) from $P(v_k \mid -) \propto P(v_k)P(\mathrm{z}_{*,k} \mid v_k, h_k, u_k)$. This belongs to a class of problems known as **sampling from doubly-intractable distributions** (Møller et al. 2006; Murray, Ghahramani, and MacKay 2006). Although it remains a challenging problem in statistics, it is not difficult to approximately sample the variables if we give up theoretical rigorousness (Liang 2010). The details of the algorithm are described in Section 5.1.

Update $\mathrm{w}_{k,*}$. The remaining problem is how to update $w_{k,m}$. Because the number of weights is very large ($K \times M$), the simple Metropolis-Hastings algorithm (Görür, Jäkel, and Rasmussen 2006; Doyle, Bicknell, and Levy 2014) is not a workable option. To address this problem, we block-sample $\mathrm{w}_{k,*} = (w_{k,1}, \cdots, w_{k,M})$ using Hamiltonian Monte Carlo (HMC) (Neal 2011). We present a sketch of the algorithm in Section 5.2.

5.1 Approximate Sampling from Doubly Intractable Distributions

During inference, we want to sample v_k (and h_k and u_k) from its posterior distribution, $P(v_k \mid -) \propto P(v_k)P(\mathrm{z}_{*,k} \mid v_k, h_k, u_k)$. Unfortunately, we cannot apply the standard Metropolis-Hastings (MH) sampler to this problem because $P(\mathrm{z}_{*,k} \mid v_k, h_k, u_k)$ contains an intractable normalization term. Such a distribution is called a **doubly intractable** distribution because Markov chain Monte Carlo itself approximates the intractable distribution (Møller et al. 2006; Murray, Ghahramani, and MacKay 2006). This problem remains an active topic in the statistics literature to date. However, if we give up theoretical rigorousness, it is not difficult to draw samples from the posterior, which are only approximately correct but work well in practice.

Specifically, we use the double MH sampler (Liang 2010). The key idea is to use an auxiliary variable to cancel out the normalization term. This sampler is based on the exchange algorithm of Murray, Ghahramani, and MacKay (2006), which samples v_k in the following steps.

1. Propose $v'_k \sim q(v'_k | v_k, \mathrm{z}_{*,k})$.
2. Generate an auxiliary variable $\mathrm{z}'_{*,k} \sim P(\mathrm{z}'_{*,k} \mid v'_k, h_k, u_k)$ using an *exact* sampler.
3. Accept v'_k with probability $\min\{1, r(v_k, v'_k, \mathrm{z}'_{*,k} \mid \mathrm{z}_{*,k})\}$, where

$$r(v_k, v'_k, \mathrm{z}'_{*,k} \mid \mathrm{z}_{*,k}) = \frac{P(v'_k)q(v_k \mid v'_k, \mathrm{z}_{*,k})}{P(v_k)q(v'_k \mid v_k, \mathrm{z}_{*,k})} \times \frac{P(\mathrm{z}_{*,k} \mid v'_k, h_k, u_k)P(\mathrm{z}'_{*,k} \mid v_k, h_k, u_k)}{P(\mathrm{z}_{*,k} \mid v_k, h_k, u_k)P(\mathrm{z}'_{*,k} \mid v'_k, h_k, u_k)} \tag{17}$$

A problem lies in the second step. The exact sampling of $\mathrm{z}'_{*,k}$ is as difficult as the original problem. The double MH sampler approximates it with a Gibbs sampling scan of $z_{l,k}$'s starting from the current $\mathrm{z}_{*,k}$. At each step of the Gibbs sampling scan, $z'_{l,k}$ is updated according to $P(z'_{l,k} \mid \mathrm{z}'_{-l,k}, v'_k, h_k, u_k)$. Note that the auxiliary variable $\mathrm{z}'_{*,k}$ is only used to compute Equation (17).

We construct the proposal distributions $q(v'_k \mid v_k, \mathrm{z}_{*,k})$ and $q(h'_k \mid h_k, \mathrm{z}_{*,k})$ using a log-normal distribution, and $q(u'_k \mid u_k, \mathrm{z}_{*,k})$ using a Gaussian distribution with mean u_k.

5.2 Hamiltonian Monte Carlo

HMC (Neal 2011) is a Markov chain Monte Carlo method for drawing samples from a probability density distribution. Unlike Metropolis-Hastings, it exploits gradient information to propose a new state, which can be distant from the current state. If no numerical error is involved, the new state proposed by HMC is accepted with probability 1.

HMC has a connection to Hamiltonian dynamics and the physical analogy is useful for gaining an intuition. In HMC, the variable to be sampled, $q \in \mathbb{R}^M$, is seen as a generalized coordinate of a system and is associated with a potential energy function $U(q) = -\log P(q)$, the negative logarithm of the (unnormalized) density function. The coordinate q is tied with an auxiliary momentum variable $p \in \mathbb{R}^M$ and a kinetic function $K(p)$. The momentum makes the object move. Because $H(q, p) = U(q) + K(p)$, the sum of the kinetic and potential energy, is constant with respect to time, the time evolution of the system is uniquely defined given an initial state (q_0, p_0). The trajectory is computed to obtain a state (q, p) at some time, and that q is the next sample we want.

Algorithm 1 HMC($U, \nabla U, q_0$).

1: $q \leftarrow q_0$
2: $p_0 \sim \mathcal{N}(\mu = 0, \Sigma = I)$
3: $p \leftarrow p_0$
4: $p \leftarrow p - \epsilon \nabla U(q)/2$
5: **for** $s \leftarrow 1, S$ **do**
6: $\quad q \leftarrow q + \epsilon p$
7: $\quad$ **if** $s < S$ **then**
8: $\quad\quad p \leftarrow p - \epsilon \nabla U(q)$
9: $\quad$ **end if**
10: **end for**
11: $p \leftarrow p - \epsilon \nabla U(q)/2$
12: $p \leftarrow -p$
13: $r \sim \text{Uniform}[0, 1]$
14: **if** $\min[1, \exp(-U(q) + U(q_0) - K(p) + K(p_0))] > r$ **then**
15: $\quad$ **return** q ▷ accept
16: **else**
17: $\quad$ **return** q_0 ▷ reject
18: **end if**

Algorithm 1 shows the pseudo-code, which is adopted from Neal (2011). The momentum variable is drawn from the Gaussian distribution (line 2). The time evolution of the system is numerically simulated using the leapfrog method (lines 4–11), where ϵ and S are parameters of the algorithm to be tuned. This is followed by a Metropolis step to correct for numerical errors (lines 13–18).

Going back to the sampling of $\mathrm{w}_{k,*}$, we need $U(\mathrm{w}_{k,*}) = -\log P(\mathrm{w}_{k,*} \mid -)$ and its gradient $\nabla U(\mathrm{w}_{k,*})$ to run HMC. The unnormalized density function $P(\mathrm{w}_{k,*} \mid -)$ is the product of (1) the probability of generating $w_{k,m}$'s from the t-distribution and (2) the probability of generating $x_{l,n}$'s for each language with $z_{l,k} = 1$. Note that $U(\mathrm{w}_{k,*})$ is differentiable because Equations (5), (6), and (7) as well as the t-distribution are differentiable.

6. Missing Value Imputation

6.1 Settings

Although our goal is to induce good latent representations, "goodness" is too subjective to be measured. To quantitatively evaluate the proposed model, we use missing value imputation as an approximate performance indicator. If the model predicts missing feature values better than reasonable baselines, we can say that the induced linguistic parameters are justified. Although no ground truth exists for the missing portion of a data set, missing value imputation can be evaluated by hiding some observed values and verifying the effectiveness of their recovery. We conducted a 10-fold cross-validation.

We ran the proposed model, now called SynDia, with two different settings: $K = 50$ and 100.[8] We performed posterior inference for 500 iterations. After that, we collected

8 In preliminary experiments, we also tried $K = 250$ and 500, but it quickly became evident that a large K caused performance degeneration.

100 samples of $x_{l,n}$ for each language, one per iteration. For each missing value $x_{l,n}$, we output the most frequent value among the 100 samples. The HMC parameters ϵ and S were set to 0.05 and 10, respectively. We applied simulated annealing to the sampling of $z_{l,k}$ and $\mathrm{x}^{\mathrm{mis}}_{l,*}$. For the first 100 iterations, the inverse temperature was increased from 0.1 to 1.0.

We also tested a simplified version of SYNDIA, SYN, from which v_k and h_k were removed. Equation (11) was replaced with Equation (14).

We compared SYN and SYNDIA with several baselines.

MFV For each feature n, always output the most frequent value among the observed portion of $\mathrm{x}_{*,n} = (x_{1,n}, \cdots, x_{L,n})$.

Surface-DIA The autologistic model applied to each surface feature n (Murawaki and Yamauchi 2018). After 500 burn-in iterations, we collected 500 samples with the interval of five iterations. Among the collected samples, we chose the most frequent feature value for each language as the output.[9]

DPMPM A Dirichlet process mixture of multinomial distributions with a truncated stick-breaking construction (Si and Reiter 2013) used by Blasi, Michaelis, and Haspelmath (2017) for missing value imputation. It assigned a single categorical latent variable to each language. As an implementation, we used the `R` package *NPBayesImpute*. We ran the model with several different values for the truncation level K^*. The best score is reported.

MCA A variant of multiple correspondence analysis (Josse et al. 2012) used by Murawaki (2015) for missing value imputation. We used the `imputeMCA` function of the `R` package *missMDA*.

MFV and Surface-DIA can be seen as models for inter-language dependencies, and DPMPM, MCA, and SYN are models for inter-feature dependencies. SYNDIA exploits both types of clues.

6.2 Results

Table 3 shows the result. We can see that SYNDIA outperformed the rest by substantial margins for both data sets. The gains of SYNDIA over SYN were statistically significant in both data sets ($p < 0.01$ for WALS and $p < 0.05$ for Autotyp). Although the likelihood $P(X \mid Z, W)$ went up as K increased, likelihood was not necessarily correlated with accuracy. Because of the high ratio of missing values, the model with a larger K can overfit the data. The best score was obtained with $K = 100$ for WALS and $K = 50$ for Autotyp, although the differences were not statistically significant in either data set. We conjecture that because WALS's features ranged over broader domains of language, a larger K was needed to cover major regular patterns in the data (Haspelmath 2008b).

The fact that SYN outperformed Surface-DIA suggests that inter-feature dependencies have more predictive power than inter-language dependencies in the data sets. However, they are complimentary in nature, as SYNDIA outperformed SYN.

DPMPM performed poorly even if a small value was set to the truncation level K^* to avoid overfitting. It divided the world's languages into a finite number of disjoint

9 Whereas Murawaki and Yamauchi (2018) partitioned $\mathrm{x}^{\mathrm{obs}}_{*,n}$ into 10 equal sized subsets for each feature n, we applied the 10-fold cross-validation to X^{obs} as a whole, in order to allow comparison with other models. At the level of features, this resulted in slightly uneven distributions over folds but did not seem to have made a notable impact on the overall performance.

Table 3
Accuracy of missing value imputation. The first column indicates the types of dependencies the models exploit: inter-language dependencies, inter-feature dependencies, or both.

Type	Model	WALS	Autotyp
Inter-language dependencies	MFV	54.80	69.42
	Surface-DIA	61.27	73.04
Inter-feature dependencies	DPMPM ($K^* = 50$)	59.12	65.53
	MCA	65.61	76.29
	SYN ($K = 50$)	72.71	81.24
	SYN ($K = 100$)	72.78	81.20
Both	SYNDIA ($K = 50$)	73.47	**81.58**
	SYNDIA ($K = 100$)	**73.57**	81.33

Table 4
Ablation experiments for missing value imputation. † and †† indicate statistically significant changes from SYNDIA with $p < 0.05$ and $p < 0.01$, respectively.

Model	WALS ($K = 100$)		Autotyp ($K = 50$)	
Full model (SYNDIA)	**73.57**		**81.58**	
-vertical	73.40	(-0.17)	81.33	(-0.25)
-horizontal	73.17	$(-0.39)^{\dagger}$	81.28	$(-0.30)^{\dagger}$
-vertical -horizontal (SYN)	72.78	$(-0.79)^{\dagger\dagger}$	81.24	$(-0.34)^{\dagger}$

groups. In other words, the latent representation of a language was a single categorical value. As expected, DPMPM showed its limited expressive power.

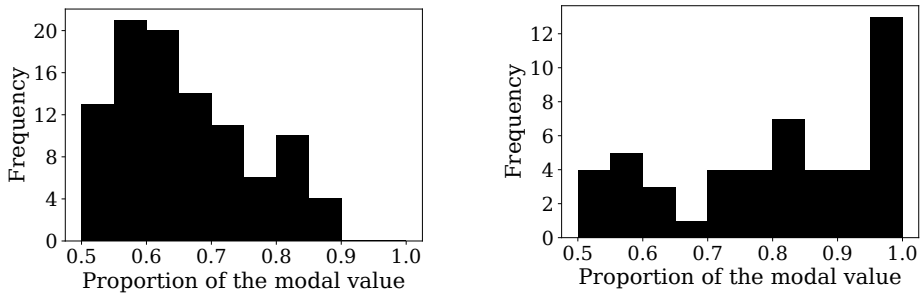
MCA uses a more expressive representation for each language: a sequence of continuous variables. It outperformed DPMPM but was inferior to SYN by a large margin. We conjecture that MCA was more sensitive to initialization than the Bayesian model armed with Markov chain Monte Carlo sampling.

To investigate the effects of inter-language dependencies, we conducted ablation experiments. We removed v_k, h_k, or both from the model. Note that if both are removed, the resultant model is SYN. The result is shown in Table 4. Although not all modifications yielded statistically significant changes, we observed exactly the same pattern in both data sets: The removal of vertical and horizontal factors consistently degenerated the performance, with h_k having larger impacts than v_k.

7. Discussion

7.1 Vertical Stability and Horizontal Diffusibility

Missing value imputation demonstrates that SYNDIA successfully captures regularities of the data. We now move on to the question of what the learned linguistic parameters



(a) WALS. (b) Autotyp.

Figure 7
The histogram of the proportions of the modal values of linguistic parameters. Since linguistic parameters are binary, the proportion is always greater than or equal to 0.5.

look like. We performed posterior inference again, with the settings based on the results of missing value imputation: We chose $K = 100$ for WALS and $K = 50$ for Autotyp. The difference was that we no longer performed a 10-fold cross-validation but gave all observed data to the models.

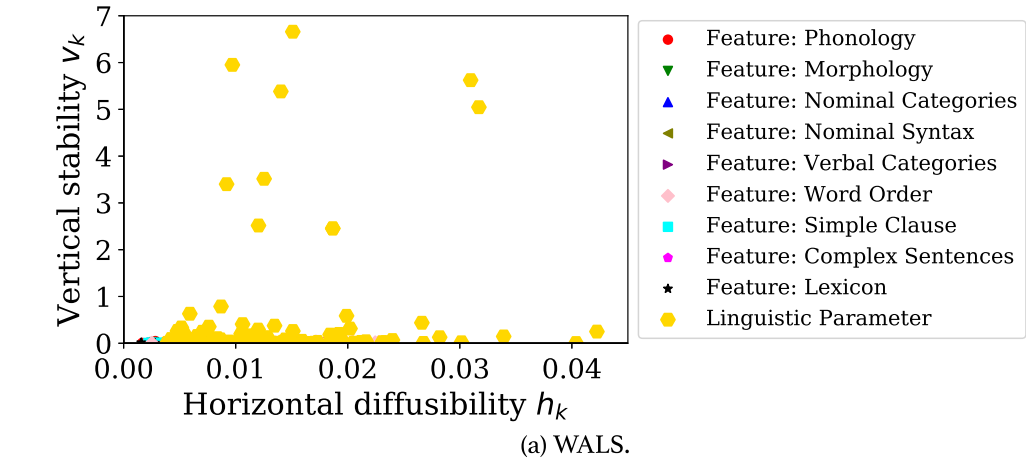
Figure 7 shows the proportion of the modal value of each linguistic parameter. For Autotyp, the largest histogram bin ranged from 0.95 to 1.0. This suggests that even with $K = 50$, macro-patterns were covered by a limited number of linguistic parameters. The rest focused on capturing micro-patterns. In contrast, linguistic parameters of WALS tended toward 0.5. There might have been more macro-patterns to be captured.

Next, we take advantage of the fact that both Surface-DIA and SYNDIA use the autologistic model to estimate vertical stability v_k (v_n for Surface-DIA) and horizontal diffusibility h_k (h_n). Large v_k indicates that phylogenetically related languages would be expected to frequently share the same value for linguistic parameter k. Similarly, h_k measures how frequently spatially close languages would be expected to share the parameter value. Because both models make use of the same neighbor graphs, we expect that if the latent representations successfully reflect surface patterns, v_k's (h_k's) are in roughly the same range as v_n's (h_n's). To put it differently, if the region in which most v_k's reside is much closer to zero than that of most v_n's, it serves as a signal of the model's failure.

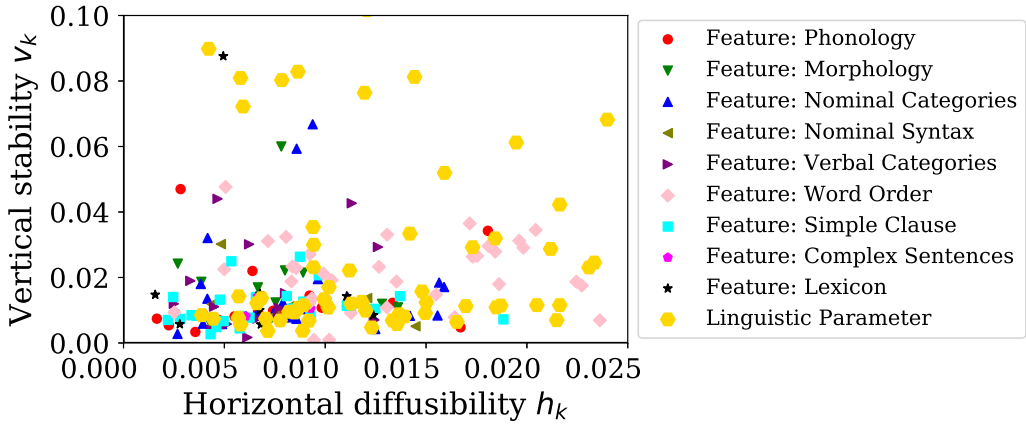
Figures 8 and 9 summarize the results. For both WALS and Autotyp, most linguistic parameters were within the same ranges as surface features, both in terms of vertical stability and horizontal diffusibility. Although this result does not directly prove that the latent representations are good, we can at least confirm that the model did not fail in this regard.

There were some outliers to be explained. Autotyp's outliers were surface features and are easier to explain: Features with very large v_n were characterized by heavy imbalances in their value distributions, and their minority values were confined to small language families.

The outliers of WALS were linguistic parameters. They had much larger v_k, much larger h_k, or both, than those of surface features, but deviations from the typical range were more marked for v_k. As we see in Figure 7(a), the linguistic parameters of WALS, including the outliers, focused on capturing macro-patterns and did not try to explain variations found in small language families. As we discuss later, these linguistic



(a) WALS.



(b) WALS (zoomed).

Figure 8
Scatter plots of the surface features and induced linguistic parameters of WALS, with vertical stability v_k (v_n) as the y-axis and horizontal diffusibility h_k (h_n) as the x-axis. Larger v_k (h_k) indicates that linguistic parameter k is more stable (diffusible). Comparing the absolute values of a v_k and an h_k makes no sense because they are tied with different neighbor graphs. Features are classified into nine broad domains (called **Area** in WALS). v_k (and h_k) is the geometric mean of the 100 samples from the posterior.

parameters were associated with clear geographical distributions, which we believe may be too clear. We are unsure whether they successfully "denoised" minor variations found in surface features or simply overgeneralized macro-patterns to missing features.

7.2 Manual Investigation of the Latent Representations

Figure 10 visualizes the weight matrix W. We can find some dashed horizontal lines, which indicate strong dependencies between feature values. However, it is difficult for humans to directly derive meaningful patterns from the noise-like images.

A more effective way to visually explore linguistic parameters is to plot them on a world map. In fact, WALS Online provides the functionality of plotting a surface

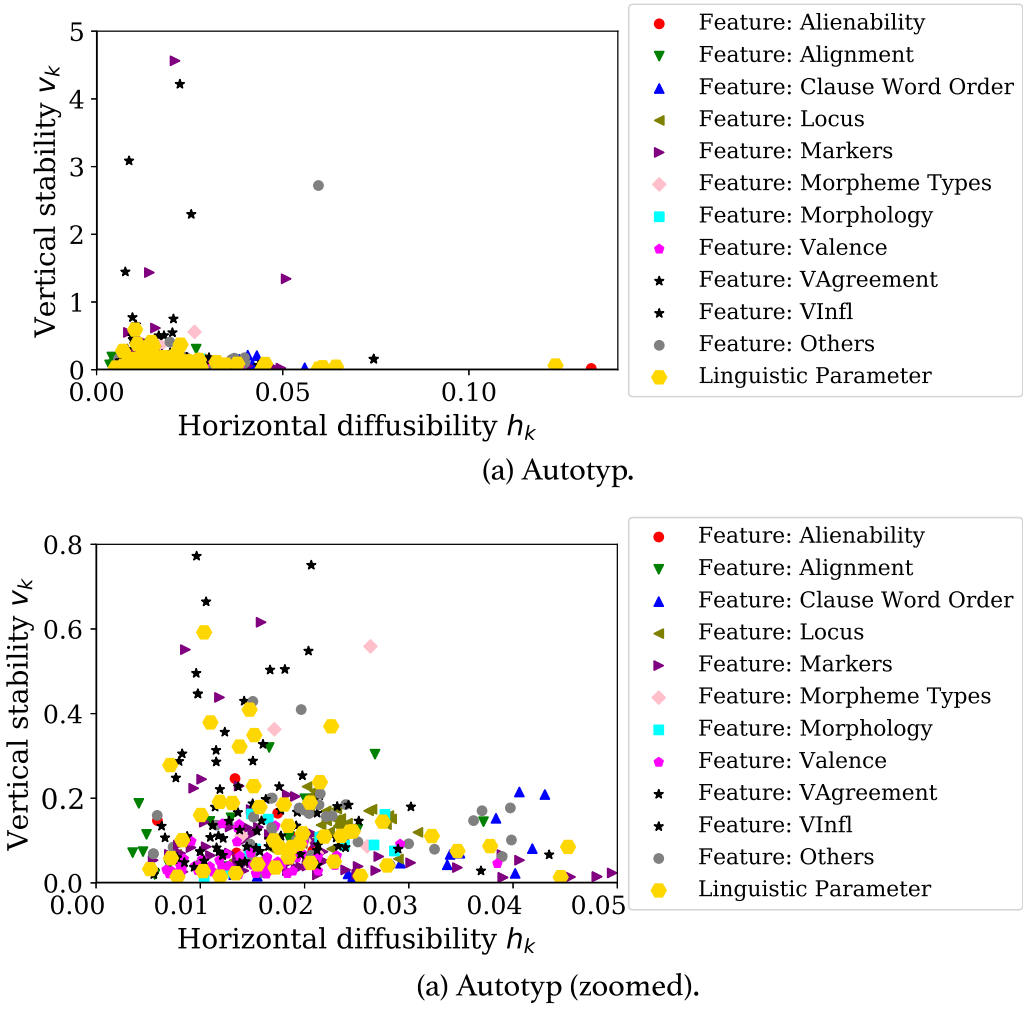
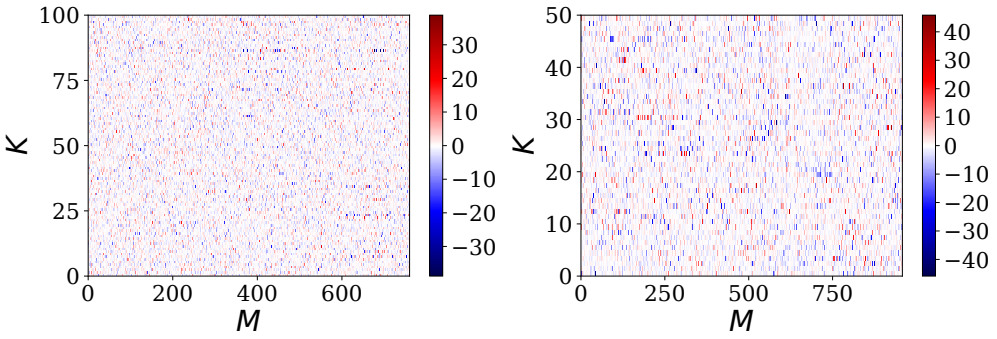


Figure 9
Scatter plots of the surface features and induced linguistic parameters of Autotyp. See Figure 8 for details.

feature and dynamically combined features to help typologists discover geographical patterns (Haspelmath et al. 2005). As exemplified by Figure 11(a), some surface features show several geographic clusters of large size, revealing something about the evolutionary history of languages. We can do the same thing for linguistic parameters.

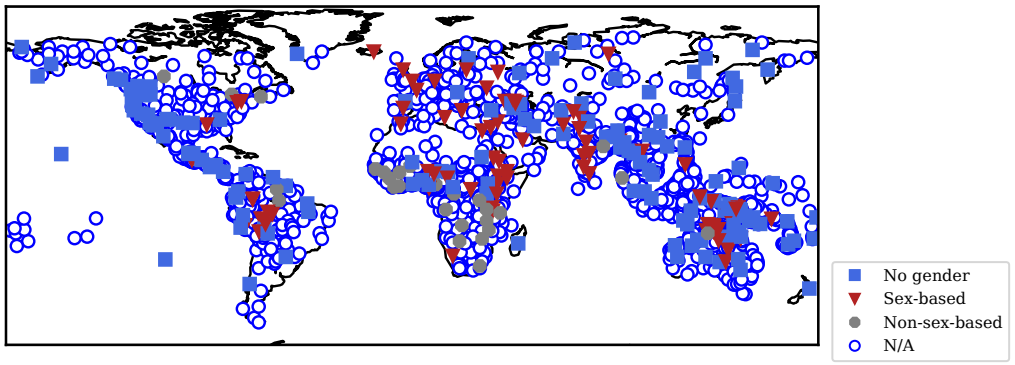
An example from WALS is shown in Figure 11(b). Even with a large number of missing values, SYNDIA yielded geographic clusters of comparable size for some linguistic parameters. Needless to say, not all surface features were associated with clear geographic patterns, and neither were linguistic parameters.

For comparison, we also investigated linguistic parameters induced by SYN (not shown in this article), which did not exploit inter-language dependencies. Some geographic clusters were found, especially when the estimation of $z_{l,k}$ was stable. In our subjective evaluation, however, SYNDIA appeared to show clearer patterns than SYN. There were many low-coverage languages, and due to inherent uncertainty, $z_{l,k}$ swung

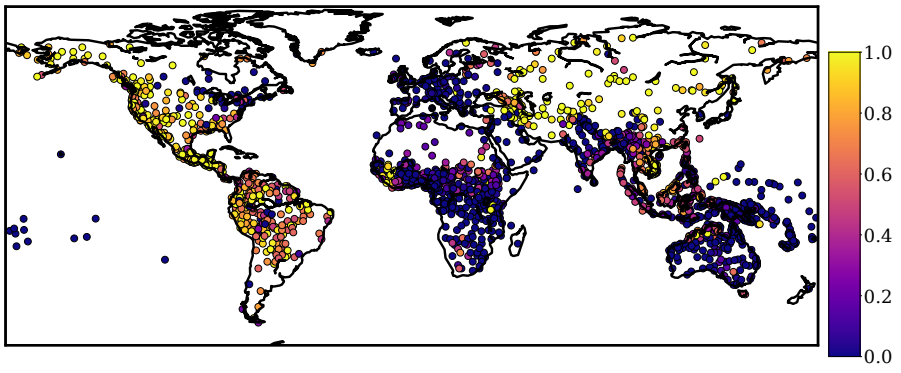


(a) WALS. (b) Autotyp.

Figure 10
Weight matrix W. Each row represents a linguistic parameter. Linguistic parameters are sorted by u_k in decreasing order. In other words, a linguistic parameter with a smaller index is more likely to be on.



(a) Feature 31A, "Sex-based and Non-sex-based Gender Systems."



(b) A linguistic parameter of SYNDIA.

Figure 11
Geographical distributions of a surface feature and SYNDIA's linguistic parameter of WALS. Each point denotes a language. For the linguistic parameter, lighter nodes indicate higher frequencies of $z_{l,k} = 1$ among the 100 samples from the posterior.

between 0 and 1 during posterior inference when inter-language dependencies were ignored. As a result, geographical signals were obscured.

The world map motivates us to look back at the cryptic weight matrix. The linguistic parameter visualized in Figure 11(b), for example, was tied to weight vector $w_{k,*}$, which gave the largest positive values to the following feature–value pairs: Feature 31A, "Sex-based and Non-sex-based Gender Systems": `No gender`; Feature 44A, "Gender Distinctions in Independent Personal Pronouns": `No gender distinctions`; Feature 30A, "Number of Genders": `None`; and Feature 32A, "Systems of Gender Assignment": `No gender`. We can determine that the main function of this linguistic parameter is to disable a system of grammatical gender. It is interesting that not all genderless languages turned on this linguistic parameter. The value `No gender` for Feature 31A is given a relatively large positive weight by another linguistic parameter, which was characterized by a tendency to avoid morphological marking. In this way, we can draw from latent parameters hypotheses of linguistic typology that are to be tested through a more rigorous analysis of data.

7.3 Applications

Although we have demonstrated in the previous section how we can directly examine the learned model for a typological inquiry, this approach has limitations. As we briefly discussed in Section 2.2, the search space must have numerous modes. A trivial way to prove this is to swap a pair of linguistic parameters together with the corresponding weight vector pair. The posterior probability remains the same. Although permutation invariance poses little problem in practice, arbitrariness involving marked–unmarked relations can be problematic. The model has a certain degree of flexibility that allows it to choose a marked form from a dichotomy it uncovers, and the rather arbitrary decision affects other linguistic parameters. The net outcome of unidentifiability is that even if the model provides a seemingly good explanation for an observed pattern, another run of the model may provide another explanation.

We suggest that instead of investing too much effort in interpreting induced linguistic parameters, we can use the model as a black box that connects two or more surface representations via the latent space. A schematic illustration of the proposed approach is shown in Figure 12. We first use the model to map a given language $x_{l,*}$ into its latent representation $z_{l,*}$, then manipulate $z_{l,*}$ in the latent space to obtain $z_{l',*}$, the latent representation of a new language l', and finally project $z_{l',*}$ back into the original space to obtain its surface representation $x_{l',*}$. By comparing $x_{l,*}$ with $x_{l',*}$, we can identify how manipulation in the latent space affects multiple surface features in general.

A useful property of this approach is that because both the surface and latent representations are discrete, linguistic parameters can readily be used as a substitute for surface features. This means that methods that have been successfully applied to surface features are applicable to linguistic parameters as well. Undoubtedly, the most promising method is Bayesian phylogenetic analysis (Dediu 2010; Greenhill et al. 2010; Dunn et al. 2011; Maurits and Griffiths 2014; Greenhill et al. 2017). For example, we can reconstruct an ancestral language from its descendants in the latent space. By comparing the ancestor and a descendant in the original space, we can investigate how languages change over time without being trapped into unnatural combinations of features, as was envisioned by Greenberg (1978). Unlike Dunn et al. (2011), who focused on the

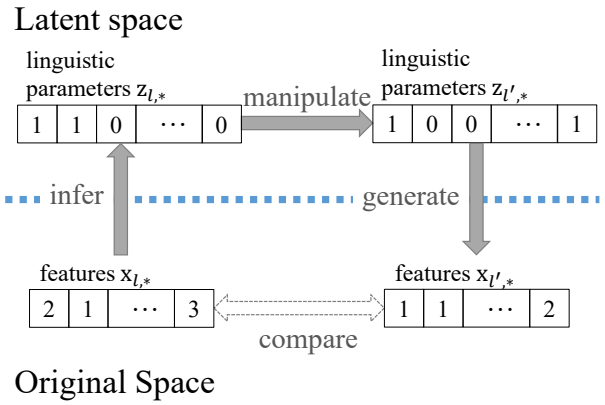


Figure 12
A schematic illustration of latent representation-based analysis.

dependency between a pair of binary features, the proposed framework has the ability to uncover correlated evolution involving multiple features (Murawaki 2018).

8. Conclusions

In this article, we reformulated the quest for Greenbergian universals with representation learning. To develop a concrete model that is robust with respect to a large number of missing values, we adopted the Bayesian learning framework. We also exploited inter-language dependencies to deal with low-coverage languages. Missing value imputation demonstrates that the proposed model successfully captures regularities of the data. We plotted languages on a world map to show that some latent variables are associated with clear geographical patterns. The source code is publicly available at `https://github.com/murawaki/lattyp`.

The most promising application of latent representation-based analysis is Bayesian phylogenetic methods. The proposed model can be used to uncover correlated evolution involving multiple features.

Acknowledgments

The key ideas and early experimental results were presented at the Eighth International Joint Conference on Natural Language Processing (Murawaki 2017). This article, which presents updated results, is a substantially extended version of the earlier conference paper. This work was partly supported by JSPS KAKENHI grant 18K18104.

References

Anderson, Stephen R. 2016. Synchronic versus diachronic explanation and the nature of the language faculty. *Annual Review of Linguistics*, 2:1–425.

Asgari, Ehsaneddin and Mohammad R. K. Mofrad. 2015. Continuous distributed representation of biological sequences for deep proteomics and genomics. *PLOS ONE*, 10(11):1–15.

Baker, Mark C. 2001. *The Atoms of Language: The Mind's Hidden Rules of Grammar*. Basic Books.

Bender, Emily M. 2016. Linguistic typology in natural language processing. *Linguistic Typology*, 20(3):645–660.

Bengio, Yoshua, Aaron Courville, and Pascal Vincent. 2013. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828.

Besag, Julian. 1974. Spatial interaction and the statistical analysis of lattice systems.

Journal of the Royal Statistical Society. Series B (Methodological), 38(2):192–236.
Bickel, Balthasar, Johanna Nichols, Taras Zakharko, Alena Witzlack-Makarevich, Kristine Hildebrandt, Michael Rießler, Lennart Bierkandt, Fernando Zúñiga, and John B. Lowe. 2017. The AUTOTYP typological databases. version 0.1.0.
Blasi, Damián E., Susanne Maria Michaelis, and Martin Haspelmath. 2017. Grammars are robustly transmitted even during the emergence of creole languages. *Nature Human Behaviour*, 1(10):723–729.
Boeckx, Cedric. 2014. What principles and parameters got wrong. In Picallo, M. Carme, editor, *Treebanks: Building and Using Parsed Corpora*, Oxford University Press, pages 155–178.
Bouckaert, Remco, Philippe Lemey, Michael Dunn, Simon J. Greenhill, Alexander V. Alekseyenko, Alexei J. Drummond, Russell D. Gray, Marc A. Suchard, and Quentin D. Atkinson. 2012. Mapping the origins and expansion of the Indo-European language family. *Science*, 337(6097):957–960.
Campbell, Lyle. 2006. Areal linguistics. In *Encyclopedia of Language and Linguistics, Second Edition*. Elsevier, pages 454–460.
Chang, Will, Chundra Cathcart, David Hall, and Andrew Garrett. 2015. Ancestry-constrained phylogenetic analysis supports the Indo-European steppe hypothesis. *Language*, 91(1):194–244.
Chen, Xi, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. 2016. InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in Neural Information Processing Systems 29*, pages 2172–2180.
Chomsky, Noam and Howard Lasnik. 1993. The theory of principles and parameters. In Joachim Jacobs, Arnim von Stechow, Wolfgang Sternefeld, and Theo Vennemann, editors, *Syntax: An International Handbook of Contemporary Research*, 1. De Gruyter, pages 506–569.
Croft, William. 2002. *Typology and Universals*. Cambridge University Press.
Cysouw, Michael. 2003. Against implicational universals. *Linguistic Typology*, 7(1):89–101.
Daumé III, Hal. 2009. Non-parametric Bayesian areal linguistics. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 593–601, Boulder, CO.
Daumé III, Hal and Lyle Campbell. 2007. A Bayesian model for discovering typological implications. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 65–72, Prague.
Dediu, Dan. 2010. A Bayesian phylogenetic approach to estimating the stability of linguistic features and the genetic biasing of tone. *Proceedings of the Royal Society of London B: Biological Sciences*, 278(1704):474–479.
Doyle, Gabriel, Klinton Bicknell, and Roger Levy. 2014. Nonparametric learning of phonological constraints in optimality theory. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1094–1103, Baltimore, MD.
Dryer, Matthew S. 1998. Why statistical universals are better than absolute universals. *Chicago Linguistic Society*, 33(2):123–145.
Dryer, Matthew S. 1989. Large linguistic areas and language sampling. *Studies in Language*, 13:257–292.
Dryer, Matthew S. 2011. The evidence for word order correlations: A response to Dunn, Greenhill, Levinson and Gray's paper in *Nature*. *Linguistic Typology*, 15:335–380.
Dunn, Michael, Simon J. Greenhill, Stephen C. Levinson, and Russell D. Gray. 2011. Evolved structure of language shows lineage-specific trends in word-order universals. *Nature*, 473(7345):79–82.
Enfield, Nicholas J. 2005. Areal linguistics and Mainland Southeast Asia. *Annual Review of Anthropology*, 34:181–206.
Gan, Zhe, Chunyuan Li, Changyou Chen, Yunchen Pu, Qinliang Su, and Lawrence Carin. 2017. Scalable Bayesian learning of recurrent neural networks for language modeling. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 321–331, Vancouver.
Goldwater, Sharon, Thomas L. Griffiths, and Mark Johnson. 2009. A Bayesian framework for word segmentation: Exploring the effects of context. *Cognition*, 112(1):21–54.
Goldwater, Sharon and Tom Griffiths. 2007. A fully Bayesian approach to unsupervised part-of-speech tagging. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 744–751, Prague.
Görür, Dilan, Frank Jäkel, and Carl Edward Rasmussen. 2006. A choice model with infinitely many latent features. In *Proceedings of the 23rd International*

Conference on Machine Learning, pages 361–368, Pittsburgh, PA.

Gray, Russell D. and Quentin D. Atkinson. 2003. Language-tree divergence times support the Anatolian theory of Indo-European origin. *Nature*, 426(6965):435–439.

Greenberg, Joseph H. 1963. Some universals of grammar with particular reference to the order of meaningful elements. In Joseph H. Greenberg, editor, *Universals of Language*, MIT Press, pages 73–113.

Greenberg, Joseph H. 1978. Diachrony, synchrony and language universals. In Joseph H. Greenberg, Charles A. Ferguson, and Edith A. Moravesik, editors, *Universals of Human Language*, volume 1. Stanford University Press, pages 61–91.

Greenhill, Simon J., Quentin D. Atkinson, Andrew Meade, and Russel D. Gray. 2010. The shape and tempo of language evolution. *Proceedings of the Royal Society B: Biological Sciences*, 277(1693):2443–2450.

Greenhill, Simon J., Chieh-Hsi Wu, Xia Hua, Michael Dunn, Stephen C. Levinson, and Russell D. Gray. 2017. Evolutionary dynamics of language systems. *Proceedings of the National Academy of Sciences, U.S.A.*, 114(42):E8822–E8829.

Griffiths, Thomas L. and Zoubin Ghahramani. 2011. The Indian buffet process: An introduction and review. *Journal of Machine Learning Research*, 12:1185–1224.

Griffiths, Thomas L., Mark Steyvers, and Joshua B. Tenenbaum. 2007. Topics in semantic representation. *Psychological Review*, 114(2):211–244.

Haspelmath, Martin. 2008a. Alienable vs. inalienable possessive constructions. Handout, Leipzig Spring School on Linguistic Diversity.

Haspelmath, Martin. 2008b, Parametric versus functional explanations of syntactic universals. In Theresa Biberauer, editor, *The Limits of Syntactic Variation*, John Benjamins, pages 75–107.

Haspelmath, Martin, Matthew Dryer, David Gil, and Bernard Comrie, editors . 2005. *The World Atlas of Language Structures*. Oxford University Press.

Heine, Bernd and Tania Kuteva. 2007. *The Genesis of Grammar: A Reconstruction*. Oxford University Press.

Hinton, Geoffrey E. 2002. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8):1771–1800.

Hinton, Geoffrey E. and Ruslan R. Salakhutdinov. 2006. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507.

Hornik, Kurt. 1991. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2):251–257.

Itoh, Yoshiaki and Sumie Ueda. 2004. The Ising model for changes in word ordering rules in natural languages. *Physica D: Nonlinear Phenomena*, 198(3):333–339.

Jasra, Ajay, Chris C. Holmes, and David A. Stephens. 2005. Markov chain Monte Carlo methods and the label switching problem in Bayesian mixture modeling. *Statistical Science*, 20(1):50–67.

Josse, Julie, Marie Chavent, Benot Liquet, and François Husson. 2012. Handling missing values with regularized iterative multiple correspondence analysis. *Journal of Classification*, 29(1):91–116.

Knowles, David and Zoubin Ghahramani. 2007. Infinite sparse factor analysis and infinite independent components analysis. In *Proceedings of the 7th International Conference on Independent Component Analysis and Signal Separation*, pages 381–388, London.

LeCun, Yann, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Liang, Faming. 2010. A double Metropolis–Hastings sampler for spatial models with intractable normalizing constants. *Journal of Statistical Computation and Simulation*, 80(9):1007–1022.

Maurits, Luke and Thomas L. Griffiths. 2014. Tracing the roots of syntax with Bayesian phylogenetics. *Proceedings of the National Academy of Sciences, U.S.A.*, 111(37):13576–13581.

Meeds, Edward, Zoubin Ghahramani, Radford Neal, and Sam Roweis. 2007. Modeling dyadic data with binary latent factors. In *Advances in Neural Information Processing Systems*, 19:977–984.

Møller, Jesper, Anthony N. Pettitt, R. Reeves, and Kasper K. Berthelsen. 2006. An efficient Markov chain Monte Carlo method for distributions with intractable normalising constants. *Biometrika*, 93(2):451–458.

Mullahy, John. 1986. Specification and testing of some modified count data models. *Journal of Econometrics*, 33(3):341–365.

Murawaki, Yugo. 2015. Continuous space representations of linguistic typology and their application to phylogenetic inference. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 324–334, Denver, CO.

Murawaki, Yugo. 2016. Statistical modeling of creole genesis. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1329–1339, San Diego, CA.

Murawaki, Yugo. 2017. Diachrony-aware induction of binary latent representations from typological features. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 451–461, Taipei.

Murawaki, Yugo. 2018. Analyzing correlated evolution of multiple features using latent representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4371–4382, Brussels.

Murawaki, Yugo and Kenji Yamauchi. 2018. A statistical model for the joint inference of vertical stability and horizontal diffusibility of typological features. *Journal of Language Evolution*, 3(1):13–25.

Murray, Iain, Zoubin Ghahramani, and David J. C. MacKay. 2006. MCMC for doubly-intractable distributions. In *Proceedings of the Twenty-Second Conference on Uncertainty in Artificial Intelligence*, pages 359–366, Cambridge, MA.

Naseem, Tahira, Regina Barzilay, and Amir Globerson. 2012. Selective sharing for multilingual dependency parsing. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 629–637, Seogwipo.

Neal, Radford M. 2011. MCMC using Hamiltonian dynamics. In Steve Brooks, Andrew Gelman, Galin L. Jones, and Xiao-Li Meng, editors, *Handbook of Markov Chain Monte Carlo*, CRC Press, pages 113–162.

Nelson-Sathi, Shijulal, Johann-Mattis List, Hans Geisler, Heiner Fangerau, Russell D. Gray, William Martin, and Tal Dagan. 2011. Networks uncover hidden lexical borrowing in Indo-European language evolution. *Proceedings of the Royal Society B: Biological Sciences*, 278:1794–1803.

Nichols, Johanna. 1992. *Linguistic Diversity in Space and Time*. University of Chicago Press.

Nichols, Johanna. 1995. Diachronically stable structural features. In Henning Andersen, editor, *Historical Linguistics 1993. Selected Papers from the 11th International Conference on Historical Linguistics, Los Angeles 16–20 August 1993*. John Benjamins Publishing Company, pages 337–355.

O'Horan, Helen, Yevgeni Berzak, Ivan Vulic, Roi Reichart, and Anna Korhonen. 2016. Survey on the use of typological information in natural language processing. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1297–1308, Osaka.

Pagel, Mark and Andrew Meade. 2006. Bayesian analysis of correlated evolution of discrete characters by reversible-jump Markov chain Monte Carlo. *American Naturalist*, 167(6):808–825.

Parkvall, Mikael. 2008. Which parts of language are the most stable? *STUF-Language Typology and Universals Sprachtypologie und Universalienforschung*, 61(3):234–250.

Pereltsvaig, Asya and Martin W. Lewis. 2015. *The Indo-European Controversy: Facts and Fallacies in Historical Linguistics*. Cambridge University Press.

Prince, Alan and Paul Smolensky. 2008. *Optimality Theory: Constraint Interaction in Generative Grammar*. John Wiley & Sons.

Si, Yajuan and Jerome P. Reiter. 2013. Nonparametric Bayesian multiple imputation for incomplete categorical variables in large-scale assessment surveys. *Journal of Educational and Behavioral Statistics*, 38(5):499–521.

Srebro, Nathan, Jason D. M. Rennie, and Tommi S. Jaakkola. 2005. Maximum-margin matrix factorization. In *Proceedings of the 17th International Conference on Neural Information Processing Systems*, pages 1329–1336, Vancouver.

Tan, Jie, John H. Hammond, Deborah A. Hogan, and Casey S. Greene. 2016. ADAGE-based integration of publicly available pseudomonas aeruginosa gene expression data with denoising autoencoders illuminates microbe-host interactions. *mSystems*, 1(1):e00025–15.

Towner, Mary C., Mark N. Grote, Jay Venti, and Monique Borgerhoff Mulder. 2012. Cultural macroevolution on neighbor graphs: Vertical and horizontal transmission among western North American Indian societies. *Human Nature*, 23(3):283–305.

Trubetzkoy, Nikolai Sergeevich. 1928. Proposition 16. In *Acts of the First International Congress of Linguists*, pages 17–18.

Tsunoda, Tasaku, Sumie Ueda, and Yoshiaki Itoh. 1995. Adpositions in word-order typology. *Linguistics*, 33(4):741–762.

Welling, Max and Yee W. Teh. 2011. Bayesian learning via stochastic gradient Langevin dynamics. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 681–688, Bellevue, WA.

Wichmann, Søren and Eric W. Holman. 2009. *Temporal Stability of Linguistic Typological Features*. Lincom Europa.