Large-sample confidence intervals of information-theoretic measures in linguistics

Abstract

This article explores a method of creating confidence bounds for information-theoretic measures in linguistics, such as entropy, Kullback-Leibler Divergence (KLD), and mutual information. We show that a useful measure of uncertainty can be derived from simple statistical principles, namely the asymptotic distribution of the maximum likelihood estimator (MLE) and the delta method. Three case studies from phonology and corpus linguistics are used to demonstrate how to apply it and examine its robustness against common violations of its assumptions in linguistics, such as insufficient sample size and non-independence of data points.

1 Introduction

Information-theoretic measures (ITMs hereafter) like entropy are commonly applied in linguistics. The most common method of estimating ITMs is based on *plug-in estimates* of probability. The estimated distribution of the random variable is taken and plugged into the formula for the *population* entropy. For instance, consider the entropy of a discrete random variable with choices $C_1, C_2, ..., C_k$. We estimate the probabilities of getting each of these choices $P(C_1), P(C_2), ..., P(C_k)$ by the *estimators* $\hat{P}(C_1), \hat{P}(C_2), ..., \hat{P}(C_k)$.[1] Then we take the formula for the entropy (1) and replace the probabilities by their estimates (2):[2]

$$\sum_{i=1}^{k} P(C_i) \ln P(C_i) \tag{1}$$

$$\sum_{i=1}^{k} \hat{P}(C_i) \ln \hat{P}(C_i) \tag{2}$$

Although it is common in linguistics to ignore the difference between (1) and (2), the difference is critical when inferential statistics is concerned. The first formula uses the 'true' probabilities, and is the unknown quantity that we estimate. The second formula uses the *estimated*

[1] The 'hat' symbol indicates that these values are the estimated values, rather than the true values.

[2] In this paper, we assume base 2 logarithms, denoted by log, in all ITMs; ln is used to denote the natural logarithm in other contexts as necessary

values of the probabilities, and it is used to calculate our estimate of the ITM. In linguistics, these estimated values are usually the proportion of data in each category.

When calculating (2), what we aim to achieve is to obtain a number close to (1). However, when linguists report the quantity calculated in (2), they typically include no estimates of uncertainty; the degree to which (2) may differ from (1) is not quantified. The problem with this approach is that we have no way of telling whether - if we gathered another set of data under identical conditions - we would get wildly different results due to sampling error alone. This would be undesirable. We should be able to find a margin of error for our estimates so that the reader knows what kind of *range* we can be reasonably sure the ITM lies in, namely with confidence intervals.

In this paper, we demonstrate how we can use a simple method from basic statistics to construct theoretically valid confidence intervals (CIs hereafter) for ITMs.[3] We also demonstrate how we can empirically validate the CIs through simulations to ensure that we have a sufficient sample size for the CIs to be reasonably accurate, and that the CIs are robust to violations of the assumptions behind the formulas that we derive for CIs. The method applies to any plug-in ITM estimate in which the estimated probabilities are maximum likelihood estimates; this allows us to also use probabilities which come from e.g. random-effect logistic regression models. Note that although our paper deals with CIs, the method can be straightforwardly modified to give Wald test p-values. We focus on ITMs on discrete random variables, as this is the most common in linguistics, but modifications for continuous distributions are also possible.

The following discussion assumes a knowledge of basic calculus, in particular partial differentiation; Supplementary Mateirals 1 provide an introduction to this topic. Those who are less familiar with symbolic calculus may still calculate the required quantities using a finite difference method provided in readily available numerical algorithm libraries, such as the R package `numDeriv` (Supplementary Materials 6), though the resulting intervals will inevitably be slightly less accurate.

2 Statistical preliminaries

We introduce the method of deriving CIs in general mathematical notation before proceeding to specific examples. This section is mainly for readers with less background in maximum likelihood estimation, which we need to derive CIs; readers familiar with the relevant statistics may skip parts of it.

2.1 Maximum likelihood estimation

The main concept behind maximum likelihood estimation is the likelihood function. Denoted by $L(\theta;X)$, it tells us the probability density of observing the data X under different values of

[3] An common family of alternative methods for deriving CIs for ITMs, the bootstrap, is discussed and contrasted with the present method in Supplementary Materials 2.

the vector of parameters θ.[4] The values that we would like to estimate are usually functions of this vector of parameters. For example, when estimating the probability that a coin will come up as heads, assuming that all successive coin flips are independent and either heads or tails,[5] the likelihood function of the function in this situation is

$$L(\theta;X_1,...,X_n) = \prod_{x=1}^{n} p^{I(X_i=1)}(1-p)^{I(X_i=0)} = p^{n_1}(1-p)^{n_0}$$

where X_i is the i-th datum, p is the probability of the result being heads, n is the sample size, n_1 and n_0 are the numbers of heads and tails, and $I(A)$ is the indicator function, i.e. $I(A) = 1$ if A is true and $I(A) = 0$ otherwise. The maximum likelihood estimate (MLE hereafter) of the parameters θ is the value of the parameter vector for which the likelihood function is maximised. Symbolically, the MLE is $\arg\max L(\theta|X)$. Usually, when finding the MLE, it is easier to maximise the log of the likelihood function. For example, to find the MLE in the coin-toss case, we may work with the log of the likelihood function as follows:

$$\ln L(\theta;X_1,...,X_n) = n_1 \ln p + n_0 \ln(1-p)$$

After differentiating this function once and equating it to 0 (this is known as the score function), the result is the MLE:

$$n_1/p - n_0/(1-p) = 0 \Rightarrow n_1(1-p) = n_0(p) \Rightarrow n_1 = p(n_1+n_0) \Rightarrow p = n_1/n$$

So the MLE for the probability of obtaining heads is simply the proportion of heads among the coin tosses. Many of the estimators commonly used by linguists are in fact MLEs, including this one.

2.2 Statistical inference with maximum likelihood estimation

An advantage of the MLE is that we know for sure how MLEs will behave *as long as there is a large enough sample size*. The properties are given by a basic theorem in statistics:

> Assume the log-likelihood function of a random variable with d-dimensional parameter vector θ is differentiable near the true value of θ and that the maximum likelihood estimator $\hat{\theta}$ exists. Subject to certain regularity conditions, we have
>
> (a) $\hat{\theta}$ converges in probability to θ and
>
> (b) $\hat{\theta}$ converges in distribution to $N(\theta, I(\theta)^{-1})$, where $I(\theta)$ is the $d \times d$ Fisher information matrix of θ with (i,j)th entry $\mathbb{E}\left[\left(\frac{\partial \ln L(\theta;X)}{\partial \theta_i}\right)\left(\frac{\partial \ln L(\theta;X)}{\partial \theta_j}\right)\right]$.

[4] The difference between the joint density function and likelihood is that in the former, the parameter is a constant and the values of the data are variables, whereas the opposite holds in the latter.

[5] Although it may feel artificial, many linguistic variables may be seen as analogous to coin tosses, such as whether a speaker will produce one variant in a syntactic or phonological alternation instead of another, or whether a language possesses some feature.

Intuitively, this tells us that a) as the sample size gets large, our estimate will be able to get arbitrarily close to the true value and that b) as long as the sample size is big enough, we can guarantee that the MLE has a normal distribution with the mean being the true value and the variance-covariance matrix being the inverse of the Fisher information.[6]

The Fisher information matrix roughly tells us how much information about the parameters we get from the data. Under mild conditions, it can be rewritten in the equivalent form $-\mathbb{E}\left[\frac{\partial^2 \ln L(\theta;X)}{\partial\theta_i \partial\theta_j}\right]$, which is more commonly used in practice. It is probably easier to understand this concept intuitively in a one-dimensional case with only one parameter, in which case the inverse of the variance of the one-dimensional normal distribution reduces to $\mathbb{E}\left[\left(\frac{\mathrm{d}\ln L(\theta;X)}{\mathrm{d}\theta}\right)^2\right]$, which is usually equivalent to $\mathbb{E}\left[-\frac{\mathrm{d}^2 \ln L(\theta;X)}{\mathrm{d}\theta^2}\right]$.

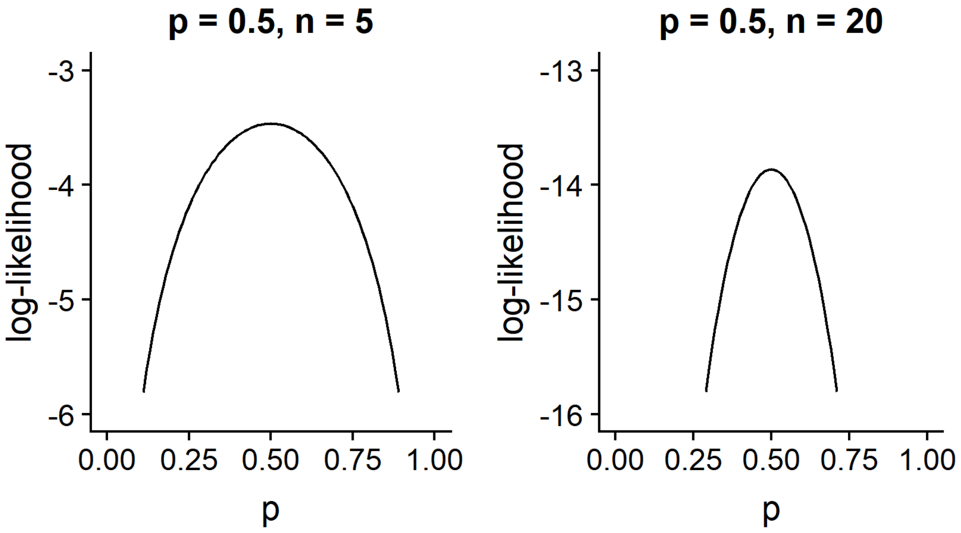
The negative second derivative of the likelihood tells us 'how concave' the likelihood function is; the greater this quantity, the 'more concave' the likelihood function is on average, and hence the more sure we can be of the estimate. If the concavity of the function is close to zero, then the likelihood function is flat, and we would expect that our estimate to be less sure and more drowned out by noise. Continuing our coin-flip example, the Fisher information is as follows:

$$\begin{aligned} -\mathbb{E}[\mathrm{d}/\mathrm{d}p\ (n_1/p - n_0/(1-p))] &= \mathbb{E}[n_1/p^2 + n_0/(1-p)^2] \\ &= np/p^2 + n(1-p)/(1-p)^2 \\ &= n/p + n/(1-p) \end{aligned} \tag{3}$$

Hence the asympotic distribution is of the MLE $\hat{p}$ is $N(p, p(1-p)/n)$, with the variance decreasing with increasing values of n.[7] To understand this intuitively, refer to the figure below, which shows the expected value of the negative log-likelihood function with $n = 5$ and $n = 20$ respectively. It's clear that the $n = 5$ graph is flatter ('less concave'), and hence we would expect the estimate to be less accurate.

[6] We leave out a lot of mathematical detail here; interested readers can consult a standard textbook in inferential statistics such as Hogg, McKean, and Craig (2005).

[7] In fact, in this simple example, we do not need to appeal to properties of the MLE; the consistency and asymptotic normality of the estimator can be derived from much more basic concepts. We simply use it for illustrative purposes.



Of course, in practice, we do not know the Fisher information matrix, since it depends on the true value of the parameter. Instead, in estimating the Fisher information matrix to derive CIs, we usually replace the parameter's value by its MLE, which we derive by calculating $I(\hat{\theta})$.[8]

Up to now, we have only discussed inference about parameters of the statistical model. However, the information-theoretic measure itself is usually not a parameter, but a *function* of the parameters, e.g. the entropy of a coin flip is a function of the parameter p. In fact, we can perform inference on ITMs in a similar way. This is because MLEs are invariant to transformations: a function of an MLE is still an MLE. Moreover, transformations of MLEs are still asymptotically normal: by the delta method (Rao, 1973, p.388), if $\phi(\theta)$ is differentiable near the true value of θ, then $\phi(\hat{\theta})$ converges in distribution to $N(\phi(\theta), D\phi(\theta)I(\theta)^{-1}D\phi(\theta)^T)$, where $D\phi(\theta)$ is the Jacobian matrix of $\phi(\theta)$.[9] From the above, we may conclude the following:

(a) $\phi(\hat{\theta})$ is a consistent estimator of $\phi(\theta)$ and

(b) $\phi(\hat{\theta})$ converges in distribution to $N(\phi(\theta), (D\phi(\theta))(I(\theta)^{-1})(D\phi(\theta))^T)$.

Let's assume that $\phi(\theta)$ is a vector of ITMs. Since we can estimate the sampling distribution of the parameter, we may construct CIs for each of its components. Since the diagonal entries of the vector of ITMs's variance-covariance matrix are the variances of estimators of the corresponding ITM, the square root of the diagonal gives the standard errors. Hence a natural Bonferroni-corrected confidence set for the function may be found as follows, using the MLE

[8] Because of this approximate nature, Wallis (2013) decries these 'Wald'-type CIs, writing that they are 'the wrong way to think about the problem' and 'we should dispense with "Wald" type approaches to confidence interval estimation'. Our position is that this method is too flexible and widely applicable to be dispensed with; rather than abandoning it, we should ensure that it works well for our purposes, as we will do in our simulation studies.

[9] This may be intuitively understood as follows: By Taylor expansion, $\phi(\hat{\theta}) \approx \phi(\theta) + \phi'(\theta)(\hat{\theta} - \theta)$ where $\hat{\theta} - \theta \sim N(0, I^{-1}(\theta))$, so we would expect $\phi(\hat{\theta})$ to be roughly distributed as $N(\phi(\theta), I^{-1}(\theta))$.

of the parameters:

$$\phi(\hat{\theta}) \pm Z_{\alpha/2g} \cdot \sqrt{diag\left(\left(D\phi(\hat{\theta})\right)\left(I(\hat{\theta})^{-1}\right)\left(D\phi(\hat{\theta})\right)^T\right)} \tag{4}$$

where g is the dimension of $\phi(\hat{\theta})$, α is the level of significance and $Z_{\alpha/2g}$ is the upper $\alpha/2g$th quantile of the standard normal distribution. For linear combinations of the components of $\phi(\hat{\theta})$, represented by $C\phi(\hat{\theta})$, we have:

$$C\phi(\hat{\theta}) \pm Z_{\alpha/2g} \cdot \sqrt{diag\left(C\left(D\phi(\hat{\theta})\right)\left(I(\hat{\theta})^{-1}\right)\left(D\phi(\hat{\theta})\right)^T C^T\right)}. \tag{5}$$

Thus, when we apply the method to a new situation, the two formulas we need to derive are a) the information matrix and b) the Jacobian matrix of the ITM, then we can derive the desired CIs. We will derive an information matrix in the next subsection, and examine concrete examples of deriving the Jacobian matrix thereafter. Note that for more complex models, either the information matrix or its inverse is usually returned by the R package, and hence there is no need to derive an explicit formula by ourselves.

2.3 Special case: Deriving the Hessian with independently and identically distributed categorical data

As we mentioned above, common practice is to use the proportion of times that a category appears in the data as the estimate of the probability of the category. Such an estimate is the maximum likelihood estimator, and can therefore be subject to the method above, under the assumption that the data are independently and identically distributed (IID) categorical variables (though we can also examine the robustness of the methods under common deviations from the ideal distribution, as we will do later). Because this method of calculating CIs is so common, we derive here the information matrix under this model assumption. Let X be the vector of n categorical data with I categories $c_1, c_2, ..., c_I$. Then its likelihood function is

$$L(\theta; X) = p_1^{n_1} p_2^{n_2} ... p_I^{n_I} \tag{6}$$

where n is the sample size, $n_i = \sum_{j=1}^{n} 1_{\{X_j = c_i\}}$ is the number of outcomes in the ith category, and p_i is the probability that a component of X belongs to the ith category, and as $p_I = 1 - \sum_{i=1}^{I-1} p_i$, the parameter vector θ only consists of $I-1$ components: $\theta = (p_1, p_2, ..., p_{I-1})^T$.

From the above, we obtain the log-likelihood function as

$$c_n(\theta) = n_1 \ln p_1 + n_2 \ln p_2 + ... + n_I \ln p_I$$

Differentiating by the parameter vector, we have

$$\frac{\partial c_n(\theta)}{\partial \theta} = \left(\frac{n_1}{p_1} - \frac{n_I}{p_I}, \frac{n_2}{p_2} - \frac{n_I}{p_I}, ..., \frac{n_{I-1}}{p_{I-1}} - \frac{n_I}{p_I}\right)^T$$

Obviously, by replacing every p_i in this expression with the plug-in estimate $\hat{p}_i = n_i/n$, every entry becomes 0. Thus the plug-in estimate is the maximum likelihood estimation under our current assumptions. In this case, the exact form of the information matrix is as follows:

$$\begin{aligned} I_{ij}(\theta) &= n\mathbb{E}\left[\left(\frac{\partial \ln L(\theta;x)}{\partial p_i}\right)\left(\frac{\partial \ln L(\theta;x)}{\partial p_j}\right)\right] \\ &= n\mathbb{E}\left[\left(\frac{1_{\{X=i\}}}{p_i} - \frac{1_{\{X=I\}}}{p_I}\right)\left(\frac{1_{\{X=j\}}}{p_j} - \frac{1_{\{X=I\}}}{p_I}\right)\right] \\ &= \begin{cases} n/p_I, & i \neq j \\ n/p_i + n/p_I, & i = j. \end{cases} \end{aligned} \tag{7}$$

Therefore, in practice, if one needs to construct a CI for an ITM, assuming the current IID categorical model, the only formula that needs to be derived is the Jacobian matrix of the ITM. We will see an example in the next subsection.

2.4 Toy example: Tone and aspiration in Lhasa Tibetan

To explain the basics of the method of deriving CIs in an easy manner, we illustrate the method introduced with a simple concrete example: we look at tone and aspiration in Tibetan. Our toy example here is selected so that readers can entertain the data even by hand to capture the basic idea behind the estimation. The CIs here are for illustrative purposes only, not tied to any specific research question. Readers familiar with the theory presented in 2.1-2.3 may skip this one, and begin reading the more realistic examples in Section 3 onwards.

We determine whether the first syllable of each word has high tone and whether it has aspiration according to the rules of Lhasa Tibetan (Tournadre & Dorje, 2003; Denwood, 1999).[10] We use the University of Virginia Spoken Tibetan Corpus (Germano, Garrett, & Weinberger, 2017). The pyewts tool (Esukhia Development Team, 2019) is used to convert the raw Tibetan text to the Extended Wylie transcription system (EWTS) (Garson & Germano, 2004) developed by the University of Virginia's Tibetan and Himalayan Library. Orthographic forms neither discussed in Tournadre and Dorje's volume nor derivable from orthographic rules were disregarded; 90 syllables (out of 737653) were discarded, which should scarcely affect the results.

In the ensuing discussion, we let p_H denote the probability that the first syllable of a Tibetan word has high tone, p_L denote the probability of low tone, p_A denote the probability of aspiration, and p_N of non-aspiration; clearly $p_H = 1 - p_L$ and $p_A = 1 - p_N$. For the joint probability of two features, we write the tone followed by the aspiration, so the probability of an aspirated,

[10] The variety spoken by Denwood's consultants seem slightly different from standard Tibetan in that the third-column stops are not aspirated; we follow the standard pronunciation, but still use Denwood's book for reference on possible orthotactic combinations. We have ignored colloquial pronunciations that diverge from the regular spelling pronunciation rules, though we acknowledge that a full analysis of Tibetan phonology should take this factor into account. For example, ཕྲུ་ གུ་ *phru.gu*, with an aspirated inital, is typically pronounced as སྤུ་ གུ་ *spu.gu*, with an unaspirated initial; in this study, we treat this word as aspirated anyway.

high-tone first syllable is p_{HA}. After tallying up the number of syllables in each category, we obtain the following table of frequencies and associated probabilities:

	aspirated	nonaspirated	total
high	66064 ($\hat{p_{HA}} = 0.0896$)	176032 ($\hat{p_{HN}} = 0.239$)	242096 ($\hat{p_H} = 0.328$)
low	137108 ($\hat{p_{LA}} = 0.186$)	358359 ($\hat{p_{LN}} = 0.486$)	495467 ($\hat{p_L} = 0.672$)
total	203172 ($\hat{p_A} = 0.275$)	534391 ($\hat{p_N} = 0.725$)	737563

Confidence interval for the entropy of tone. We start with a confidence interval for the entropy of tone. To make things simple, we first ignore the information provided by aspiration (see the next example for the case considering both features). This only involves one parameter (the parameter of a high tone) and one ITM (the entropy of tone), so matrix notation is not needed; the situation is in effect that of our coin flip example. (Using the notation of section 2.3, p_H is p_1 and p_L is p_2 with $I = 2$.) Then the entropy of tone is $H_T = -p_H \log(p_H) - p_L \log(p_L)$, where $p_L = 1 - p_H$. Therefore, by plugging in, we have $\hat{H_T} = -\hat{p_H} \log(\hat{p_H}) - \hat{p_L} \log(\hat{p_L}) = -0.328 \log 0.328 - 0.672 \log 0.672 = 0.9131157$.

Using the product rule and changing to base e logarithms, we have

$$\mathrm{d}H_T/\mathrm{d}p_H = (\ln 2)^{-1}(-\ln(p_H) - 1 + \ln(p_L) + 1) = (\ln 2)^{-1}(\log(p_L) - \log(p_H))$$

Therefore, our estimate of the derivative is as follows:

$$\mathrm{d}\hat{H_T}/\mathrm{d}\hat{p_H}|_{\hat{p_H}=0.328} = (\ln 2)^{-1}(\ln(1-\hat{p_H}) - \ln(\hat{p_H}))|_{\hat{p_H}=0.328} = (\ln 2)^{-1}(\ln 0.672 - \ln 0.328) = 1.03321$$

Thus, we may estimate the variance of $\hat{H_T}$ as follows:

$$\hat{Var}(\hat{H_T}) = (\ln 2)^{-2}(\ln(\hat{p_L}) - \ln(\hat{p_H}))^2 (n/\hat{p_H} + n/\hat{p_L})^{-1} = 3.191 \times 10^{-7}$$

where the $(n/p_H + n/p_L)^{-1}$ comes from (3). Our required confidence interval is therefore $\hat{H_T} \pm 1.96\sqrt{\hat{Var}(\hat{H_T})} = (0.912, 0.914)$.

Simultaneous confidence intervals for the entropies of tone and aspiration. The above example is very simple, containing only one parameter. Our second example make of the matrix notation introduced in the subsection before, as we will begin to consider the entropies of tone and aspiration simultaneously.

To evaluate this, we consider the joint distribution of tone and aspiration. The parameter vector for this distribution has three components: $(p_{HA}, p_{HN}, p_{LA})^T$. (Using the notation in section 2.3, $p_{HA} = p_1, p_{HN} = p_2, p_{LA} = p_3, p_{LN} = p_4$ and $I = 4$.) The value of the information-theoretic measure $\phi(\theta)$ and the Fisher information matrix $I(\theta)$ are as follows:

$$\phi(\theta) = \begin{bmatrix} -p_H \log(p_H) - p_L \log(p_L) \\ -p_A \log(p_A) - p_N \log(p_N) \end{bmatrix}, I(\theta) = \begin{bmatrix} n/p_{HA} + n/p_{LN} & n/p_{LN} & n/p_{LN} \\ n/p_{LN} & n/p_{HN} + n/p_{LN} & n/p_{LN} \\ n/p_{LN} & n/p_{LN} & n/p_{LA} + n/p_{LN} \end{bmatrix}$$

and hence their respective estimates are

$$\phi(\hat{\theta}) = \begin{bmatrix} 0.9131157 \\ 0.8491962 \end{bmatrix}$$

and

$$I(\hat{\theta}) = \begin{bmatrix} n/\hat{p_{HA}} + n/\hat{p_{LN}} & n/\hat{p_{LN}} & n/\hat{p_{LN}} \\ n/\hat{p_{LN}} & n/\hat{p_{HN}} + n/\hat{p_{LN}} & n/\hat{p_{LN}} \\ n/\hat{p_{LN}} & n/\hat{p_{LN}} & n/\hat{p_{LA}} + n/\hat{p_{LN}} \end{bmatrix}$$

$$= \begin{bmatrix} 737563/0.0896 + 737563/0.486 & 737563/0.486 & 737563/0.486 \\ 737563/0.486 & 737563/0.239 + 737563/0.486 & 737563/0.486 \\ 737563/0.486 & 737563/0.486 & 737563/0.186 + 737563/0.486 \end{bmatrix}$$

$$= \begin{bmatrix} 9.752 \times 10^6 & 1.518 \times 10^6 & 1.518 \times 10^6 \\ 1.518 \times 10^6 & 4.608 \times 10^6 & 1.518 \times 10^6 \\ 1.518 \times 10^6 & 1.518 \times 10^6 & 5.486 \times 10^6 \end{bmatrix}.$$

The six entries of the Jacobian matrix are computed similarly as the above example. When performing the differentiations, bear in mind that p_L is treated as $1 - p_H = 1 - p_{HA} - p_{HN}$ and *not* $p_{LA} + p_{LN}$, since p_{LN} is not part of the parameter vector and cannot be used. Similarly, p_N is treated as $1 - p_A = 1 - p_{HA} - p_{LA}$ and not as $p_{HN} + p_{LN}$:

$$D\phi(\theta)$$

$$= \begin{bmatrix} \frac{\partial - p_H \log(p_H) - p_L \log(p_L)}{\partial p_{HA}} & \frac{\partial - p_H \log(p_H) - p_L \log(p_L)}{\partial p_{HN}} & \frac{\partial - p_H \log(p_H) - p_L \log(p_L)}{\partial p_{LA}} \\ \frac{\partial - p_A \log(p_A) - p_N \log(p_N)}{\partial p_{HA}} & \frac{\partial - p_A \log(p_A) - p_N \log(p_N)}{\partial p_{HN}} & \frac{\partial - p_A \log(p_A) - p_N \log(p_N)}{\partial p_{LA}} \end{bmatrix}$$

$$= \frac{1}{\ln 2} \begin{bmatrix} -\ln p_H - 1 + \ln p_L + 1 & -\ln p_H - 1 + \ln p_L + 1 & 0 \\ -\ln p_A - 1 + \ln p_N + 1 & 0 & -\ln p_A - 1 + \ln p_N + 1 \end{bmatrix}$$

$$= \frac{1}{\ln 2} \begin{bmatrix} -\ln p_H + \ln p_L & -\ln p_H + \ln p_L & 0 \\ -\ln p_A + \ln p_N & 0 & -\ln p_A + \ln p_N \end{bmatrix}$$

Hence the estimated Jacobian matrix is calculated as follows:

$$D\phi(\hat{\theta}) = \frac{1}{\ln 2} \begin{bmatrix} -\ln \hat{p_H} + \ln \hat{p_L} & -\ln \hat{\hat{p_H}} + \ln \hat{p_L} & 0 \\ -\ln \hat{p_A} + \ln \hat{p_N} & 0 & -\ln \hat{p_A} + \ln \hat{p_N} \end{bmatrix} = \begin{bmatrix} 1.033 & 1.033 & 0 \\ 1.395 & 0 & 1.395 \end{bmatrix}$$

Using the results above, the ITMs' estimated variance-covariance matrix is calculated as follows:

$$\hat{Var}(\phi(\hat{\theta}))$$

$$= D\phi(\hat{\theta})(I(\hat{\theta}))^{-1}(D\phi(\hat{\theta}))^T$$

$$= \begin{bmatrix} 1.033 & 1.033 & 0 \\ 1.395 & 0 & 1.395 \end{bmatrix} \begin{bmatrix} 9.752 \times 10^6 & 1.518 \times 10^6 & 1.518 \times 10^6 \\ 1.518 \times 10^6 & 4.608 \times 10^6 & 1.518 \times 10^6 \\ 1.518 \times 10^6 & 1.518 \times 10^6 & 5.486 \times 10^6 \end{bmatrix}^{-1} \begin{bmatrix} 1.033 & 1.395 \\ 1.033 & 0 \\ 0 & 1.395 \end{bmatrix}$$

$$= \begin{bmatrix} 3.191 \times 10^{-7} & -1.655 \times 10^{-9} \\ -1.655 \times 10^{-9} & 5.267 \times 10^{-7} \end{bmatrix}$$

Using this estimated covariance matrix, we can now calculate the 95% confidence intervals for our two ITMs. Since we have two ITMs, we need to correct for multiple comparisons. We use the Bonferroni correction here; since there are two, we require each CI to have 97.5% confidence level. Thus we use the upper and lower $0.05/(2 \cdot 2) = 0.0125$ quantiles of the standard normal distribution as critical values, i.e. ± 2.241 resulting in CIs of $0.913 \pm 2.241(\sqrt{3.191 \times 10^{-7}}) = (0.912, 0.914)$ and $0849 \pm 2.241(\sqrt{5.267 \times 10^{-7}}) = (0.848, 0.851)$.

The reader may now be thinking that matrix notation is not so useful, since we can simply derive CIs separately for the two, and the result would be the same. In fact it is possible to use the covariance to give an exact 95% confidence ellipse rather than two Bonferroni-corrected intervals (which would have greater than 95% asymptotic confidence level). Although statistically more efficient, this method is more difficult to interpret especially when the confidence region has over three dimensions, so we do not follow this approach. We will now demonstrate the value of matrix notation using a case where the covariances are useful.

Confidence interval for the difference in entropy between tone and aspiration. In the above examples, we have a very large sample, so would expect the CIs for the two ITMs to be very narrow. However, the same probably cannot be said if we take the difference in entropy between tone and aspiration, because the difference is small enough that even a narrow confidence interval may include 0. The difference is estimated at $\hat{H_T} - \hat{H_A} = 0.0639195$. Given how small the difference is, it is less intuitive how confident we can be that tone has higher entropy.

To resolve the issue, we take the variance-covariance matrix of the two ITMs from the previous subection and derive the variance of their difference from it. By taking $C = \begin{bmatrix} 1 & -1 \end{bmatrix}$, the difference between the two entropies may be represented as $C\phi(\hat{\theta})$, and we have

$$\hat{Var}(C\phi(\hat{\theta})) = C\hat{Var}(\phi(\hat{\theta}))C^T = \begin{bmatrix} 1 & -1 \end{bmatrix} \begin{bmatrix} 3.191 \times 10^{-7} & -1.655 \times 10^{-9} \\ -1.655 \times 10^{-9} & 5.267 \times 10^{-7} \end{bmatrix} \begin{bmatrix} 1 \\ -1 \end{bmatrix} = 8.567 \times 10^{-7}$$

Our required confidence interval is $0.0639195 \pm 1.96 \times \sqrt{8.567 \times 10^{-7}} = (0.0621, 0.0657)$. This strongly suggests that tone has higher entropy than aspiration.

2.5 Bootstrap methods as an alternative

Before we proceed to actual applications of the method, we briefly introduce a good alternative that can be used if the normal confidence intervls are not working well - that is, the point estimates are too biased, the standard errors estimates are too low, and/or the sampling distribution is too far removed from the normal distribution, whether because of limited sample size or violations against assumptions of the statistical model assumed. The method is known as the bootstrap. There is a substantial literature on this technique, and we only go over the basics here.

The idea behind the bootstrap is to create new samples from the dataset that we have obtained, and use the sampling distribution of the parameter estimates calculated using the resampled 'fake' datasets to infer the sampling distribution of the parameter estimate using the real dataset. There are two main types of bootstrap: Parametric bootstrap generates new data from the statistical model that we assume using the parameter estimates from the real data; nonparametric bootstrap generates new data by resampling from the actual real data.

The simplest kind of bootstrap, based on assuming that the data follow an IID categorical distribution, is to simply resample all data points with replacement, calculate the ITMs in each iteration, and somehow use the distribution of these ‘fake’ ITMs to draw inference about the true value of the ITM. (In this case, the parametric and nonparametric bootstrap methods are the same.) The basic bootstrap interval is one way of doing this. The idea behind basic bootstrap intervals is as follows: We assume that the bootstrap distribution $\phi(\hat{\theta}^*) - \phi(\hat{\theta})$ roughly approximates the true distribution $\phi(\hat{\theta}) - \phi(\theta)$. We may find a rough $(1-\alpha)\%$ confidence interval corrected for bias by using $(\phi(\hat{\theta}) - \varepsilon^*_{1-\alpha/2}, \phi(\hat{\theta}) - \varepsilon^*_{\alpha/2})$, where $\varepsilon^*_{\alpha/2}$ is the upper $\alpha/2$-th quantile of the distribution the ‘residuals’, with the b-th residual ε^*_b calculated as $\phi(\hat{\theta}^*_b) - \phi(\hat{\theta})$. The calculation of the basic interval may be simplified as $(2\phi(\hat{\theta}) - \phi(\hat{\theta}^*)_{1-\alpha/2}, 2\phi(\hat{\theta}) - \phi(\hat{\theta}^*)_{\alpha/2}$, where $\phi(\hat{\theta}^*)_{\alpha/2}$ refers to the upper $\alpha/2$-th quantile of the distribution of bootstrap estimates of $\phi(\theta)$. Therefore, the basic bootstrap method can handle several problems with normal confidence intervals that may arise because of limited sample size: it can (a) correct our estimate of the ITM for bias, (b) give us a better estimate of the standard error by simulation, and (c) dispense with the normality asusmption.

If even the basic bootstrap is insufficient, it is likely better to choose a bootstrap method with faster convergence, such as bootstrap-*t* (Efron & Tibshirani, 1994), the iterated bootstrap (S. M. S. Lee & Young, 1995), or both. Bootstrap-*t* requires an estimate of standard error in each iteration of the bootstrap; we may either use the Hessian for estimation (though this may be inaccurate since, as we have seen before, with small sample sizes the standard errors may be underestimated) or do another layer of bootstrap on the first-order bootstrap data. (Some other bootstrap methods do not correct for bias, and thus are not appropriate here.)

As we shall see, bootstrap methods constitute a viable alternative to normal confidence intervals under certain situations where the normal intervals perform poorly.

Now that we have introduced the method of deriving CIs with our toy example, we are moving on to more realistic case studies: (a) marginal entropies and functional loads of the components of the Cantonese syllable, (b) the use of the Kullback-Leibler divergence (KLD) for model assessment in an artificial language learning experiment, and (c) the use of mutual information and conditional entropy to explore the syntax-lexicon interface in Classical Chinese.

3 Case study 1: Difference between marginal entropies and functional loads of syllable components in Cantonese

Cantonese syllables maximally contain four components: onset, nucleus, coda and tone. Do and Lai (2019) examined the relative contribution of different syllable components in determining perceptual distance between Cantonese syllables, and took an information-theoretic approach to explaining the differences. The proposed explanation was that the greater the uncertainty within a component, the greater discriminating power it has, and hence the heavier its weight in determining phonological distance.

Entropy and functional load measures were made using data from the Hong Kong Cantonese Corpus (Luke & Wong, 2015) extracted with the PyCantonese interface (J. L. Lee, Chen, & Tsui, 2016), and their plug-in estimates taken. The entropies of onsets, nuclei, codas and tones were estimated to be 3.882, 2.866, 2.659 and 2.529 respectively. It is unclear, however, whether the differences between them, particularly the last three, are simply due to sampling error. We thus construct CIs for each of the six possible pairings between the four syllable components (onset vs nucleus, onset vs coda, etc.).

One could derive CIs separately for four components and examine their overlap, but this ignores the dependency between components. For example, given that the coda is only among {/p/, /t/, /k/}, the probability of the tone being among {2, 4, 5} would be zero since such combinations are banned by phonotactics (Bauer & Benedict, 1997). We thus derive intervals for the differences directly.

3.1 Proposal

Confidence interval for the entropy. Let the datum $X_j = (o_j, n_j, c_j, t_j)^T$ be the jth syllable token sampled from Cantonese speech, represented by a four-dimensional random vector of syllable components. Each attested combination of the value of the four components is a syllable type. Suppose there are A onsets $o_1, o_2, ..., o_A$, B nuclei $n_1, n_2, ..., n_B$, C codas $c_1, c_2, ..., c_C$ and D tones $t_1, t_2, ..., t_D$ making up I possible syllables $s_1, s_2, ..., s_I$ with $s_i = (o_{a_i}, n_{b_i}, c_{c_i}, t_{d_i})^T$. In the $o_{a_i}, n_{b_i}, c_{c_i}$ and t_{d_i}, the a_ith onset is the onset of the ith syllable and so on. Assuming the data follow a categorical distribution with parameter vector $\theta = (p_1, p_2, ..., p_{I-1})^T$, the entropies of the four syllable components are thus:

$$H(\theta) = \Bigg(-\sum_{a=1}^{A} P(o = o_a) \log P(o = o_a), -\sum_{b=1}^{B} P(n = n_a) \log P(n = n_a), \\ -\sum_{c=1}^{C} P(c = c_c) \log P(c = o_c), -\sum_{d=1}^{D} P(t = t_a) \log P(t = t_a) \Bigg)^T \quad (8)$$

To see how this depends on the parameter vector, let $S = \{s_1, s_2, ..., s_I\}$ and $S(o_a) = \{s_i \in S : o_{a_i} = o_a\}$, i.e. the set of syllables for which the onset is o_a. $S(n_b), S(c_c)$ and $S(t_d)$ are defined analogously. Then we have

$$P(o = o_a) = \sum_{k | s_k \in S(o_a)} p_k, P(n = n_b) = \sum_{k | s_k \in S(n_b)} p_k, P(c = c_c) = \sum_{k | s_k \in S(c_c)} p_k, P(t = t_d) = \sum_{k | s_k \in S(t_d)} p_k.$$

We now consider the value of $\dfrac{\partial - \sum_{a=1}^{A} P(o = o_a) \log P(o = o_a)}{\partial p_i}$; results of the other three components are similar. Note that only two terms in the summation depend on p_i, namely the ones corresponding to the onsets of the ith and Ith syllables (since p_I depends on p_i).

We first assume that the Ith and ith syllable do not share an onset. Then for the term with the probability of the ith syllable we have $\frac{\partial P(o=o_{a_i})\log P(o=o_{a_i})}{\partial p_i} = (\ln 2)^{-1}(\ln P(o=o_{a_i})p_i + 1)$, and for the other term we have $\frac{\partial P(o=o_{a_I})\log P(o=o_{a_I})}{\partial p_i} = (\ln 2)^{-1}(-\ln(P(o=o_{a_I})-1)$. Thus $\frac{\partial - \sum_{a=1}^{A} P(o=o_a)\log P(o=o_a)}{\partial p_i} = \log(P(o=o_{a_I})) - \log(P(o=o_{a_i}))$.

Now assume that the Ith and ith syllable do share an onset. Note that $P(o=o_{a_i}) = \sum_{s_k\in S(o_a)} p_k = \sum_{s_k\in S(o_a),k\neq I} p_k + 1 - \sum_{i=1}^{I} p_i$, which is free of p_i because p_i has coefficient +1 in the first summation and -1 in the second. Thus $\frac{-\partial \sum_{a=1}^{A} P(o=o_a)\log P(o=o_a)}{\partial p_i} = 0$. We may understand this as follows: the probability of the onset of a syllable having the same value as the last syllable can be obtained by subtracting the probabilities of the other values from 1, and the other values do not depend on p_i.

Therefore, the (j,i)th entry of the Jacobian matrix is the log-probability of the jth syllable component having the same value as the ith syllable minus the log-probability of the jth syllable component having the same value as the Ith syllable. The case when the ith syllable has the same jth syllable component as the Ith syllable can be subsumed under this description since $\log(P(o=o_{a_I})) - \log(P(o=o_{a_i})) = \log(P(o=o_{a_I})) - \log(P(o=o_{a_I})) = 0$.

From the data, we obtain following estimate of the covariance matrix:

$$D\phi(\hat{\theta})(I(\hat{\theta})^{-1})D\phi(\hat{\theta})^T$$
$$= \begin{bmatrix} 6.863\times 10^{-6} & 8.199\times 10^{-7} & -2.029\times 10^{-7} & 6.039\times 10^{-7} \\ 8.199\times 10^{-7} & 7.88\times 10^{-6} & -8.329\times 10^{-7} & 7.62\times 10^{-7} \\ -2.029\times 10^{-7} & -8.329\times 10^{-7} & 9.161\times 10^{-6} & -5.55\times 10^{-7} \\ 6.039\times 10^{-7} & 7.62\times 10^{-7} & -5.55\times 10^{-7} & 1.179\times 10^{-6} \end{bmatrix}$$

To find confidence bounds for the differences, we take the matrix

$$C = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 1 & 0 & -1 & 0 \\ 1 & 0 & 0 & -1 \\ 0 & 1 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{bmatrix}$$

which gives us

$$C\phi(\hat{\theta}) = (1.016, 1.224, 1.353, 0.2076, 0.3369, 0.1293)^T \text{ and } CD\phi(\hat{\theta})(I(\hat{\theta})^{-1})D\phi(\hat{\theta})^T C^T$$
$$= \begin{bmatrix} 1.31\times 10^{-5} & 5.413\times 10^{-6} & 6.201\times 10^{-6} & -7.69\times 10^{-6} & -6.902\times 10^{-6} & 7.881\times 10^{-7} \\ 5.413\times 10^{-6} & 1.643\times 10^{-5} & 5.907\times 10^{-6} & 1.102\times 10^{-5} & 4.938\times 10^{-7} & -1.052\times 10^{-5} \\ 6.201\times 10^{-6} & 5.907\times 10^{-6} & 6.835\times 10^{-6} & -2.942\times 10^{-7} & 6.333\times 10^{-7} & 9.276\times 10^{-7} \\ -7.69\times 10^{-6} & 1.102\times 10^{-5} & -2.942\times 10^{-7} & 1.871\times 10^{-5} & 7.396\times 10^{-6} & -1.131\times 10^{-5} \\ -6.902\times 10^{-6} & 4.938\times 10^{-7} & 6.333\times 10^{-7} & 7.396\times 10^{-6} & 7.535\times 10^{-6} & 1.395\times 10^{-7} \\ 7.881\times 10^{-7} & -1.052\times 10^{-5} & 9.276\times 10^{-7} & -1.131\times 10^{-5} & 1.395\times 10^{-7} & 1.145\times 10^{-5} \end{bmatrix}$$

With $g = 6$ and $\alpha = 0.05$, we have $\alpha/2g = .05/12 = 0.004166667$ and $z_{0.004166667} = 2.638257$. From these, we obtain the CIs (1.007, 1.026), (1.213, 1.234), (1.346, 1.36), (0.1962, 0.219), (0.3297, 0.3441), (0.1203, 0.1382) for the entropy difference between onsets and nuclei, onsets and codas, onsets and tones, nuclei and codas, nuclei and tones, and codas and tones respectively. Since the Bonferroni procedure is conservative, if the estimated asymptotic normal approximations are good, we would have over 95% confidence that the true differences lie in the intervals, strongly supporting that the differences exist.

Confidence interval for the functional load. The formula for functional load is as follows:

$$FL(\mathbf{C}) = \frac{H(L) - H(L'_{\mathbf{C}})}{H(L)} \tag{9}$$

where L refers to the actual language, here Cantonese, and $L'_{\mathbf{C}}$ refers to a fictional state of the language in which all contrasts in the syllable component $\mathbf{C} \in \{\mathbf{o}, \mathbf{n}, \mathbf{c}, \mathbf{t}\}$ are neutralised.

Note that we have

$$H(L) = -\sum_{i=1}^{I} P(s = c_i) \log P(s = c_i)$$

and hence $\frac{\partial H(L)}{\partial p_i} = \log(p_I) - \log(p_i)$.

Similarly, we have

$$H(L'_{\mathbf{C}}) = -\sum_{i=1}^{I'_{\mathbf{C}}} P_{\mathbf{C}}'(s \in s'_{i\mathbf{C}}) \log P_{\mathbf{C}}'(s \in s'_{i\mathbf{C}})$$

where $P'_{\mathbf{C}}(s = c_i)$ is the probability of a syllable being c_i under the fictional language state with $\mathbf{C}$ neutralised, and $s_{\mathbf{C}}{}'_1, s_{\mathbf{C}}{}'_2, ..., s_{\mathbf{C}}{}'_{I'}$ are sets of syllables in the original language that belong to the $I_{\mathbf{C}}'$. Despite the complicated notation, a moment's reflection reveals that the derivative of this entropy is also quite simple: $\frac{\partial H(L)}{\partial p_i} = \log(p_{I\mathbf{C}}') - \log(p_{i\mathbf{C}}')$ where $p_{i\mathbf{C}}'$ is the probability that a syllable is the same as c_i under the modified language. By the quotient rule, we can thus calculate the derivative of the functional load as follows:

$$\frac{\partial FL(L_{\mathbf{C}})}{\partial p_i} = \frac{(\log(p_I) - \log(p_i) - \log(p_{I\mathbf{C}}') + \log(p_{i\mathbf{C}}'))(H(L)) - (H(L) - H(L'_{\mathbf{C}}))(\log(p_I) - \log(p_i))}{(H(L))^2}$$

This time we have the vector of functional loads $(0.1763, 0.09028, 0.0804, 0.1015)^T$ and the covariance matrix

$$D\phi(\hat{\theta})(I(\hat{\theta})^{-1})D\phi(\hat{\theta})^T = \begin{bmatrix} 1.959 \times 10^{-7} & 3.251 \times 10^{-8} & 3.895 \times 10^{-8} & 4.715 \times 10^{-8} \\ 3.251 \times 10^{-8} & 1.082 \times 10^{-7} & 2.729 \times 10^{-8} & 2.479 \times 10^{-8} \\ 3.895 \times 10^{-8} & 2.729 \times 10^{-8} & 1.266 \times 10^{-7} & 2.526 \times 10^{-8} \\ 4.715 \times 10^{-8} & 2.479 \times 10^{-8} & 2.526 \times 10^{-8} & 1.336 \times 10^{-7} \end{bmatrix}$$

Using the same $\mathbf{C}$, we have $C\phi(\hat{\theta}) = (0.086, 0.09588, 0.07477, 0.009878, -0.01123, -0.02111)^T$ and $CD\phi(\hat{\theta})(I(\hat{\theta})^{-1})D\phi(\hat{\theta})^T C^T$

$$= \begin{bmatrix} 2.39\times10^{-7} & 1.517\times10^{-7} & 1.41\times10^{-7} & -8.731\times10^{-8} & -9.8\times10^{-8} & -1.069\times10^{-8} \\ 1.517\times10^{-7} & 2.446\times10^{-7} & 1.35\times10^{-7} & 9.29\times10^{-8} & -1.667\times10^{-8} & -1.096\times10^{-7} \\ 1.41\times10^{-7} & 1.35\times10^{-7} & 2.352\times10^{-7} & -5.977\times10^{-9} & 9.418\times10^{-8} & 1.002\times10^{-7} \\ -8.731\times10^{-8} & 9.29\times10^{-8} & -5.977\times10^{-9} & 1.802\times10^{-7} & 8.133\times10^{-8} & -9.888\times10^{-8} \\ -9.8\times10^{-8} & -1.667\times10^{-8} & 9.418\times10^{-8} & 8.133\times10^{-8} & 1.922\times10^{-7} & 1.108\times10^{-7} \\ -1.069\times10^{-8} & -1.096\times10^{-7} & 1.002\times10^{-7} & -9.888\times10^{-8} & 1.108\times10^{-7} & 2.097\times10^{-7} \end{bmatrix}$$

and the CIs are (0.08471, 0.08729), (0.09458, 0.09718), (0.07349, 0.07605), (0.008758, 0.011), (-0.01239, -0.01008), (-0.02232, -0.0199).

3.2 Simulations

Simulation under ideal conditions. Having derived the formula, we now validate it by simulation for two reasons. First, we must check for errors in our formula derivation or code. Secondly, the theory on which our formula is based only works for large sample sizes, so we much make sure the current size suffices.

The idea behind the simulations is to generate *fake* corpora from the assumed underlying model, then compute the empirical distributions of the entropy and functional load differences to ensure that they closely match the theoretical ones we assume. In particular, we check if (a) the mean of the entropy and functional load difference estimates is close to the true values, (b) the covariance matrix of the estimates is close to the theoretically expected one, and (c) the distribution is normal.

For our simulation study, we generate 1000 fake corpora same size as HKCanCor, taking the empirical probabilities from HKCanCor as the true underlying probabilities. Therefore, the values of $C\phi(\hat{\theta})$ and $CD\phi(\hat{\theta})(I(\hat{\theta})^{-1})D\phi(\hat{\theta})^T C^T$ from the last section are the values of $C\phi(\theta)$ and $CD\phi(\theta)(I(\theta)^{-1})D\phi(\theta)^T C^T$ in this simulation respectively. Looking at the distribution of the point estimates of the entropy differences, the mean is $(1.016, 1.224, 1.353, 0.2077, 0.3368, 0.1291)^T$ while sample covariance matrix is as follows:

$$\begin{bmatrix} 1.324\times10^{-5} & 4.528\times10^{-6} & 6.012\times10^{-6} & -8.713\times10^{-6} & -7.229\times10^{-6} & 1.484\times10^{-6} \\ 4.528\times10^{-6} & 1.653\times10^{-5} & 5.381\times10^{-6} & 1.2\times10^{-5} & 8.528\times10^{-7} & -1.115\times10^{-5} \\ 6.012\times10^{-6} & 5.381\times10^{-6} & 6.612\times10^{-6} & -6.311\times10^{-7} & 6\times10^{-7} & 1.231\times10^{-6} \\ -8.713\times10^{-6} & 1.2\times10^{-5} & -6.311\times10^{-7} & 2.071\times10^{-5} & 8.082\times10^{-6} & -1.263\times10^{-5} \\ -7.229\times10^{-6} & 8.528\times10^{-7} & 6\times10^{-7} & 8.082\times10^{-6} & 7.829\times10^{-6} & -2.529\times10^{-7} \\ 1.484\times10^{-6} & -1.115\times10^{-5} & 1.231\times10^{-6} & -1.263\times10^{-5} & -2.529\times10^{-7} & 1.238\times10^{-5} \end{bmatrix}$$

which closely resemble the theoretical mean vector and covariance matrix from the previous section. Finally, the Shapiro-Wilk test shows that the distributions of each of the individual estimates are marginally univariate normal ($p = 0.2835, 0.4872, 0.5397, 0.2004, 0.1707, 0.5654$).[11]

We also compute CIs for the differences between the entropy values for each fake corpus. The overall confidence level for the entire set is 96.4%, fairly close to the nominal level of 95%. The coverages for each difference are 0.989, 0.993, 0.993, 0.994, 0.989, 0.991 respectively.

[11] We do not use multivariate normality in our CIs, so it is not necessary to check multivariate normality.

Similarly, for functional loads, the mean vector of the difference estimates is $(0.9859, 0.9315, 0.7722, 0.4137, 0.02825, 0.07925)^T$, the covariance matrix is

$$\begin{bmatrix} 2.602\times10^{-7} & 1.692\times10^{-7} & 1.591\times10^{-7} & -9.103\times10^{-8} & -1.011\times10^{-7} & -1.008\times10^{-8} \\ 1.692\times10^{-7} & 2.524\times10^{-7} & 1.447\times10^{-7} & 8.327\times10^{-8} & -2.442\times10^{-8} & -1.077\times10^{-7} \\ 1.591\times10^{-7} & 1.447\times10^{-7} & 2.435\times10^{-7} & -1.434\times10^{-8} & 8.439\times10^{-8} & 9.873\times10^{-8} \\ -9.103\times10^{-8} & 8.327\times10^{-8} & -1.434\times10^{-8} & 1.743\times10^{-7} & 7.669\times10^{-8} & -9.761\times10^{-8} \\ -1.011\times10^{-7} & -2.442\times10^{-8} & 8.439\times10^{-8} & 7.669\times10^{-8} & 1.855\times10^{-7} & 1.088\times10^{-7} \\ -1.008\times10^{-8} & -1.077\times10^{-7} & 9.873\times10^{-8} & -9.761\times10^{-8} & 1.088\times10^{-7} & 2.064\times10^{-7} \end{bmatrix}$$

and normality is still not rejected except for the second last difference (p = 0.9859, 0.9315, 0.7722, 0.4137, 0.02825, 0.07925). Fortunately, the deviation from normality is not serious (Supplementary Materials 3).

This time, the overall confidence level is only 30.1%. For each pair of differences, the confidence levels are $(1, 1, 1, 0.932, 0.827, 0.328)^T$, so we may be more suspicious about the last difference, but otherwise the results are fairly trustworthy. Moreover, because the mean of the estimates is actually *closer* to zero than the true value, this lack of coverage likely does not affect our conclusions, so there is no need to be suspicious of our results.

Simulation under realistic conditions. In reality, however, most corpus data are not truly independent as we assume in our underlying model. For example, a syllable is more likely to be followed by syllables with which it forms a frequent word (e.g. *hoi1* is more frequently followed by *sam1* to form *hoi1sam1* 'happy' than a random syllable like *sang1*). To test the method under this condition, we compute the estimated probability of each *word* appearing in the corpus and simulate our fake corpora from this probability distribution over words rather than characters. We then repeate our character-based CIs analysis. The difference is negligible. Results are reported in Supplementary Materials 3.

4 Case study 2: Kullback-Leibler divergence and model assessment

In artificial language learning, we often test hypotheses about learning by conducting experimental studies and comparing the results against computational learning models corresponding to different hypotheses. The results of learning are a discrete probability distribution. One may evaluate how far the computational models diverge from experimental data using Kullback-Leibler divergence (KLD). The model with greater KLD can be said to perform worse. However, given that error occurs in experimental data, it is not the ground truth itself, but merely a reflection of it contaminated by sampling error. It is thus important to examine whether the difference in KLD is due to real differences between the quality of the models or merely noise.

We take Albright and Do (2013) as an example. In their artificial language learning experiment, each datum produced by each participant may fall into three categories: A voicing alternation, a continuancy alternation, and no alternation. In the experiment, participants were either exposed more to the voicing alternation than the continuancy alternation (Language 1) or vice versa (Language 2). Moreover, the alternation occurred with both coronals and labials, with coronals appearing more often. The alternation rate, ignoring the no-alternation condition, was analysed using a Bernoulli mixed-effects logistic regression model with place of articulation and language as factors; the place-language interaction was found to be insignificant and therefore dropped.

Several constraint-based models of language learning, with varying hypotheses on learning biases, were trained and compared to empirical data. These models encode roughly the same types of constraints, differing only in priors set on constraint weightings and the presence of a general constraint. They were trained until the percentage of no alternation matched that of the experimental data as closely as possible. Albright and Do computed the KLD between model predictions and empirical data, and found that the most complex model with all biases present performed best. Their KLD was computed using the raw percentages from the experiments and the models' probability outputs, and were averaged out across the four conditions.

We take a slightly different approach here: Instead of considering all three response categories, we ignore the no-alternation category, since the models were trained specifically to match the empirical non-alternation rate, and hence the numerical difference between the emprical and predicted rates is not really of interest. In other words, unlike in the original study, where the variable of interest (for which KLDs were calculated) was the consonant status of the phonological output, our variable of interest is the consonant status of the output *given that an alternation occurred.*

Moreover, for our estimate of 'ground truth', instead of raw percentages, we will take the predictions from the parsimonious logistic regression model. Since the learning models predict no between-item or between-subject differences in probabilities, we will take the probabilities assuming by-item and by-subject random effects are zero. This means taking the median probability (across items and subjects) predicted by the logistic regression model.

4.1 Proposal

This situation is not complicated mathematically, but the large amount of quantities involved makes the notation confusing. Therefore, we must clarify all notation used. We denote the four learning models by M_1, M_2, M_3, M_4 and the logistic regression model by M_{LR}. Let $p_{M_i,Pl,Lg}$ denote the probability that the alternation more frequent in the training data is produced, i.e. voicing for Language 1 and continuancy for Language 2, for model M_i, place of articulation $Pl \in \{\text{cor}, \text{lab}\}$ and language Lg. Note that the probabilities for the four learning models are *known* whereas the probabilities $p_{M_{LR},Pl,Lg}$ are unknown and estimated by $\hat{p}_{M_{LR},Pl,Lg}$. Finally, denote by the random variable X_{Lg,Pl,M_i} the type of alternation produced for language Lg and place of articulation Pl by model M_i.

Now, the logistic regression model is as follows, with the parameter vector $\beta = (\beta_0, \beta_1, \beta_2)^T$:

$$\log(\frac{p}{1-p}) = (\beta_0 + b_0) + (\beta_1 + b_1)X_1 + \beta_2 X_2, (b_0, b_1)^T \sim N(0, \Sigma) \tag{10}$$

where, since sum coding rather than dummy coding is employed, $X_1 = 1$ for coronals and -1 for labials, and $X_2 = -1$ for Language 1 and -1 for Language 2, i.e.

$$p = \frac{1}{\exp(-(\beta_0 + b_0) - (\beta_1 + b_1)X_1 - \beta_2 X_2) + 1}.$$

Hence, taking the average of the random effects, we have

$$\begin{cases} p_{M_{lr},\text{cor},Lg_1} = \dfrac{1}{\exp(-\beta_0 - \beta_1 + \beta_2) + 1} \\ p_{M_{lr},\text{lab},Lg_1} = \dfrac{1}{\exp(-\beta_0 + \beta_1 + \beta_2) + 1} \\ p_{M_{lr},\text{cor},Lg_2} = \dfrac{1}{\exp(-\beta_0 - \beta_1 - \beta_2) + 1} \\ p_{M_{lr},\text{lab},Lg_2} = \dfrac{1}{\exp(-\beta_0 + \beta_1 - \beta_2) + 1} \end{cases}$$

Let $p_{M_{lr}}(\beta)$ denote the vector $(p_{M_{lr},\text{cor},Lg_1}, p_{M_{lr},\text{lab},Lg_1}, p_{M_{lr},\text{cor},Lg_2}, p_{M_{lr},\text{lab},Lg_2})^T$ and p_{M_i} denote the vector $(p_{M_i,\text{cor},Lg_1}, p_{M_i,\text{lab},Lg_1}, p_{M_i,\text{cor},Lg_2}, p_{M_i,\text{lab},Lg_2})^T$. Denote by $D_{Pl,Lg}(X_{M_i}||X_{M_{lr}})$ the KLD between the logistic regression prediction and learning model prediction for the distribution of alternations for place Pl and language Lg. We have

$$D_{Pl,Lg}(X_{M_i}||X_{M_{lr}}) = p_{M_i,Pl,Lg} \log \frac{p_{M_i,Pl,Lg}}{p_{M_{lr},Pl,Lg}} + (1 - p_{M_i,Pl,Lg}) \log \frac{1 - p_{M_i,Pl,Lg}}{1 - p_{M_{lr},Pl,Lg}}$$

Finally, denote $D_{M_i} = (D_{Cor,Lg_1}(X_{M_i}||X_{M_{lr}}), D_{Lab,Lg_1}(X_{M_i}||X_{M_{lr}}), D_{Cor,Lg_2}(X_{M_i}||X_{M_{lr}}), D_{Lab,Lg_2}(X_{M_i}||X_{M_{lr}}))$ and denote the concatenation of the KLD vectors for the four models as the 16-dimensional vector $D(p_{M_{lr}}) = (D_{M_1}^T, D_{M_2}^T, D_{M_3}^T, D_{M_4}^T)^T$.

The attentive reader may have noticed that we have made the dependence of D on $p_{M_{lr}}$ and of $p_{M_{lr}}$ on β explicit. This is to help us with the differentiation. It may be clumsy to differentiate the KLDs directly with respect to the beta's. However, by the multivariable chain rule, the Jacobian matrix of the KLDs, which we may denote as $DD \circ p$, can be calculated by first computing the Jacobian matrices DD and $Dp_{M_{lr}}$ individually, then multiplying them together.

Let us start by computing $Dp_{M_{lr}}$. Let $\text{sgn}_{Pl,Lg}(i)$ denote the sign of β_i in the argument of the exponential function in $p_{M_{lr},Pl,Lg}$. (For the two sum-coded variables, this is the *opposite* sign as in the sum coding.) Then, by the chain rule, we have

$$\frac{\partial p_{M_{lr},Pl,Lg}}{\partial \beta_i} = \frac{-\text{sgn}_{Pl,Lg}(i)\exp(\text{sgn}_{Pl,Lg}(0)\beta_0 + \text{sgn}_{Pl,Lg}(1)\beta_1 + \text{sgn}_{Pl,Lg}(2)\beta_2)}{\exp(\text{sgn}_{Pl,Lg}(0)\beta_0 + \text{sgn}_{Pl,Lg}(1)\beta_1 + \text{sgn}_{Pl,Lg}(2)\beta_2) + 1}$$

Now let's consider DD. This is very straightforward: when the KLD is differentiated with respect to a probability from a different condition, the derivative is 0, and when it is differentiated with respect to the probability from the same condition, the derivative is

$$\frac{\partial D_{Pl,Lg}(X_{M_i}||X_{M_{lr}})}{p_{M_{lr},Pl,Lg}} = \frac{1}{\ln 2}\left(-\frac{p_{M_i,Pl,Lg}}{p_{M_{lr},Pl,Lg}} + \frac{1 - p_{M_i,Pl,Lg}}{1 - p_{M_{lr},Pl,Lg}}\right)$$

Thus, the distribution of the sixteen KLDs is estimated as follows:

$$V\hat{a}r(D) = (D(D \circ p_{M_{lr}})(\hat{\beta}))V\hat{a}r(\hat{\beta})(D(D \circ p_{M_{lr}})(\hat{\beta}))^T$$

$\hat{Var}(\hat{\beta})$ is obtained using the `vcov` function in `lme4` (Bates, Maechler, Bolker, Walker, et al., 2015). The resulting 16×16 matrix is too large to print here.

Finally, we compute five different contrasts: Model 1 vs Model 2, Model 1 vs Model 3, Model 2 vs Model 3. Model 2 vs Model 4 and Model 3 vs Model 4.[12] The coefficient of the model with the higher index is positive, so if both limits of the CI are below 0, then we may conclude that the model with higher index is *better*. As we average out across the four conditions, the KLDs from the model with the larger index in the comparison receive a weight of 0.25, and those with the smaller index receive a weight of -0.25.

The estimated vector of differences $C\hat{D}$ was calculated as $(-0.2323, -0.1501, 0.08223, 0.2745, 0.1922)^T$ and the estimated covariance matrix was

$$\begin{bmatrix} 0.004499 & 0.00577 & 0.001271 & 0.002219 & 0.0009473 \\ 0.00577 & 0.01104 & 0.005267 & 0.01027 & 0.005006 \\ 0.001271 & 0.005267 & 0.003996 & 0.008055 & 0.004059 \\ 0.002219 & 0.01027 & 0.008055 & 0.01627 & 0.008211 \\ 0.0009473 & 0.005006 & 0.004059 & 0.008211 & 0.004152 \end{bmatrix}$$

and the resulting CIs are (-0.4051, -0.05956), (-0.4207, 0.1205), (-0.08059, 0.2451), (-0.05406, 0.603) and (0.02624, 0.3582). We may conclude that Model 2 outperforms Model 1, but most other comparisons are inconclusive, and in fact Model 3 outperforms Model 4. In fact, the sign of three of the five point estimates is positive.

Why is this the case? It turns out that $\hat{p}_{lr}$ is quite different from the probabilities estimated from raw percentages (used in the original paper). We have $\hat{p}_{lr} = (0.9648, 0.5673, 0.8849, 0.2690)^T$, whereas from the raw percentages (ignoring inter-speaker variation), we have $\hat{p}_{raw} = (0.7990, 0.7365, 0.5310, 0.3731)^T$. Why is this the case? Looking at individual-level estimates of the probability of the more frequent alternation given the presence of an alternation, the estimated median probabilities are $(1.000, 0.500, 1.000, 0.250)^T$. Thus the 'average' participant's probabilities seem better estimated by the probabilities from the logistic regression model than the raw percentages. For Language 1 coronals, for example, the outliers that greatly disfavour the more frequent alternation bring down the estimated $\hat{p}_{raw}$ substantially, whereas $\hat{p}_{lr}$ is much less affected.

4.2 Simulation

We use the same method of validating the CIs: treating the estimated parameter values as 'true' values, then simulating from them. The function `simulate.merMod` in `lme4` is used to create 1000 simulated datasets of 'fake data' using the parameter estimates. The option `use.u` is set to false, so that random effects are generated at each iteration instead of taken from the estimates. The KLD differences estimated from each simulated dataset have a mean of $(-0.2068, 0.005436, 0.2122, 0.5376, 0.3254)^T$, and the covariance matrix of the estimates is as follows:

[12] This way, all models are compared with models equal or one step away in complexity.

$$\begin{bmatrix} 0.006403 & 0.01416 & 0.007762 & 0.0153 & 0.007539 \\ 0.01416 & 0.05416 & 0.03999 & 0.08042 & 0.04043 \\ 0.007762 & 0.03999 & 0.03223 & 0.06512 & 0.03289 \\ 0.0153 & 0.08042 & 0.06512 & 0.1316 & 0.06648 \\ 0.007539 & 0.04043 & 0.03289 & 0.06648 & 0.03359 \end{bmatrix}$$

Presumably due to the much smaller sample size ($n \approx 621$), the simulation results are not as neat as in Case Study 1. The KLD estimates are highly biased (except the first difference), and the estimated covariance matrix and empirical covariance matrix do not match up very well, especially from the second KLD difference onwards, where our asymptotic method substantially underestimates the standard error. The normal assumption was not rejected (p = 0.9426, 0.8459, 0.9118, 0.1994, 0.9119, 0.1383). The empirical coverage is 34.7% overall and and 99.3%, 93.1%, 35.9%, 36.8% and 43.2% for the individual differences. The first difference (where we find a 'significant' difference between the two KLDs) seems relatively accurate.

A possible solution, to further test the 'remaining' difference hypothesis, is to use bootstrap methods. Using the same simulated datasets as parametric bootstrap resamples, basic confidence intervals (Davison & Hinkley, 1997) of the KLD differences were computed as $(-0.4634, -0.145), (-0.5559, 0.2045), (-0.135, 0.3936), (-0.1704, 0.9004), (-0.0312, 0.5079)$. Here, the final difference includes zero, and thus the only conclusion we may draw us that Model 2 outperforms Model 1.

We performed simulations for the bootstrap method as well. Because of the computational intensiveness of the operation, we only chose the first 100 simulated datasets from the simulation of the normal confidence intervals. The resulting empirical confidence levels were 89%, 82%, 78%, 77%, 76% for each separate difference, 71% overall. Although not quite impressive, this represents a substantial improvement over the normal confidence intervals.

Note that in our simulation, we have ignored one source of variability: The number of non-alternation items is assumed fixed. A more realistic simulation would bring in variability in this respect, perhaps with a multinomial logistic regression model or adding a binary logistic regression model predicting whether an alternation will occur.

5 Case study 3: Conditional entropy, mutual information and the syntax-lexicon interface in Classical Chinese

Thus far, we have only compared different ITMs from the same sample, e.g. the entropy of different syllable components in Case Study 1 or the KLDs of the same empirically-determined distribution against different model predictions in Case Study 2. In this section, we will investigate a 'two independent samples' problem, i.e. comparing results across two datasets to determine whether a difference in ITMs exists, with a case study of Classical Chinese negators.

In Old Chinese, two preverbal negators, *bù* 不 and *fú* 弗, perform similar functions, perhaps with an earlier aspectual distinction (Pulleyblank, 2010). *Fú* later declined in usage and died out in modern standard Chinese, whereas *bù* remains in use (Liu, 2004). Our interest in this section is to examine the diachronic devleopment of *bù* and *fú*. We want to investigate two questions:

- Has the predictability of the *bù/fú* alternation from head verb increased between Warring States and Hàn texts?

- Does *fú* take a narrower range of verbs compared to *bù*, and has the difference widened between the two periods?[13]

The texts we compare are the *Zuǒ Zhuàn* 左傳 and *Gǔliáng Zhuàn* 穀梁傳. Both are annotations of the *Spring and Autumn Annals*, and hence similar enough in genre to compare. The *Zuǒ Zhuàn* was written during the Warring States era, whereas the *Gǔliáng Zhuàn* passed down orally at first before appearing in written form during the Hàn Dynasty. We would therefore expect the latter work to reflect language use in the Hàn Dynasty (Gu, 1998). Both works are available online on the online database Wikisource.[14] We exported the Wikisource pages for these works, extracted all clauses with the two negators, and determined the verb modified by the negator in each clause; details of the annotation scheme are given in Supplementary Materials 5.

To address the first question, we take the mutual information between the verb and the negator of each clause, normalised by dividing it by the negator's entropy. The resulting quantity may be interpreted as how much of the uncertainty in the negator overlaps with the uncertainty in verb choice; if the verb completely determines the negator, this normalised mutual information measure is 1, whereas if the verb tells us nothing about the negator, then normalised mutual information is 0. We then find the difference between normalised MIs in the two books. The entropy of the negator is estimated at 0.4720281 for the *Zuǒ Zhuàn* and 0.3117744 for the *Gǔliáng Zhuàn*. The MI is estimated at 0.1926379 for the *Zuǒ Zhuàn* and 0.2235097 for the *Gǔliáng Zhuàn*. The two normalised MIs are thus 0.4081069 and 0.7168955, with an estimated difference of 0.3087887. Considering that the normalised MIs fall between 0 and 1, a 0.3 difference seems very substantial.

To address the second question, we use pointwise conditional entropy. That is, we find the amount of uncertainty in the choice of verbs given that the negator is *bù* and given that the negator is *fú*. We first examine whether the conditional entropy given *bù* is greater than given *fù* in the two samples, and if so, whether the gap has widened between the two periods - which we would expect if *fú* has become more restricted in what it modifies. The conditional entropies for *bù* and *fú* are 7.590992 and 5.382707 (difference: 2.208284) for the *Zuǒ Zhuàn* and 6.273054 and 2.991033 (difference: 3.282021) for the *Gǔliáng Zhuàn*. The difference between the differences is 1.073737. Numerically, given that the negator is *fú*, the verb seems substantially more predictable than if the negator is *bù*, and this difference has increased between the Warring States and Hàn eras.

[13] For the convenience of discussion, we hypothesise fú to take narrower scope than bù, but for the purpose of calculation, it should not matter.

[14] http://zh.wikisource.org/

5.1 Proposal

Confidence interval for the normalised mutual information. The situation is similar to the Cantonese syllable case, so we reuse the notation from there. Let the datum X_j be the jth negator-verb pair sampled. Each datum is a random vector $X_j = (n_j, v_j)$. Each attested combination of the possible values of the two components is a clause type. There are two negators n_1, n_2, and suppose there are C verbs $v_1, v_2, ..., v_C$.

Further suppose there are I attested clause types $c_1, c_2, ..., c_I$ with $c_i = (n_{a_i}, v_{b_i})^T$, i.e. the a_ith negator is the negator of the ith negative clause type and so on. Then under the assumption that the data follow an IID categorical distribution with parameter vector $\theta = (p_1, p_2, ..., p_{I-1})^T$, the mutual information between negator and verb is as follows:

$$\begin{aligned} I_{n;v}(\theta) &= \sum_{i=1}^{I} P(n = n_{a_i}, v = v_{b_i}) \log\left(\frac{P(n = n_{a_i}, v = v_{b_i})}{P(n = n_{a_i})P(v = v_{b_i})}\right) \\ &= \sum_{i=1}^{I} P(c = c_i) \log\left(\frac{P(c = c_i)}{P(n = n_{a_i})P(v = v_{b_i})}\right) \end{aligned} \tag{11}$$

and the formula for the negators' entropy is as follows:

$$H_n(\theta) = -P(n = n_{a_i}) \log P(n = n_{a_i}) \tag{12}$$

Again, to see how this formula depends on the parameters, let $C = c_1, c_2, ..., c_I$ and $C(n_i) = \{c_i \in C : n_{a_i} = n_a\}$, i.e. the set of syllables with onset n_a. $C(v_b)$ is defined analogously. As before, we have

$$P(n = n_a) = \sum_{k|c_k \in S(n_a)} p_k, P(v = v_b) = \sum_{k|c_k \in S(v_b)} p_k.$$

To save space, we will often omit the random variables in our probabilities, e.g. $P(n = n_a)$ will be written as $P(n_a)$. The partial derivatives of $H_n(\theta)$ are calculated as in Case Study 1: the derivative is $(\ln 2)^{-1}(\ln P(n_{a_i}) - P(n_{a_I}))$. We focus on the partial derivatives of mutual information.

We first consider differentiating the mutual information with respect to parameters corresponding to the probability of clauses sharing neither negator nor verb with c_I. Let p_k be the probability by which we are differentiating. We decompose each summand in the summation in the formula of MI into the sum of three terms (according to the three probabilities inside the log sign), and differentiate each term individually. The results are as follows:

Term	Derivative
$\sum_{i=1}^{I} P(c_i) \log P(c_i)$	$\frac{1}{\ln 2}(\ln P(c_k) + 1 - \ln P(c_I) - 1)$
$-\sum_{i=1}^{I} P(c_i) \log P(n_{a_i})$	$-\frac{1}{\ln 2}\left(\ln P(n_{a_k}) + \sum_{i\mid a_i = a_k} \frac{P(c_i)}{P(n_{a_k})} - \ln P(n_{a_I}) - \sum_{i\mid a_i = a_I} \frac{P(c_i)}{P(n_{a_I})}\right)$
$-\sum_{i=1}^{I} P(c_i) \log P(v_{b_i})$	$-\frac{1}{\ln 2}\left(\ln P(v_{b_k}) + \sum_{i\mid b_i = b_k} \frac{P(c_i)}{P(v_{b_k})} - \ln P(v_{b_I}) - \sum_{i\mid b_i = b_I} \frac{P(c_i)}{P(v_{b_I})}\right)$

Clearly the 1 and -1 of the first row cancel out. By the theorem of total probability, the four summations in the second and third rows all add up to 1 too, so they also cancel out. Thus the derivative of the MI is $\log P(c_k) - \log P(n_{a_k}) - \log P(n_{b_k}) - \log P(c_k) + \log P(n_{a_I}) + \log P(n_{b_I})$ or, perhaps more intuitively, $\log\left(\frac{P(c_k)}{P(n_{a_k})P(v_{b_k})}\right) - \log\left(\frac{P(c_I)}{P(n_{a_I})P(v_{b_I})}\right)$.

Now let's consider the case where the negator-clause combination corresponding to p_k has the same negator as the last item, i.e. $n_{a_k} = n_{a_I}$:

Term	Derivative
$\sum_{i=1}^{I} P(c_i) \log P(c_i)$	$\frac{1}{\ln 2}(\ln P(c_k) + 1 - \ln P(c_I) - 1)$
$-\sum_{i=1}^{I} P(c_i) \log P(n_{a_i})$	0
$-\sum_{i=1}^{I} P(c_i) \log P(v_{b_i})$	$-\frac{1}{\ln 2}\left(\ln P(v_{b_k}) + \sum_{i\mid b_i=b_k} \frac{P(c_i)}{P(v_{b_k})} - \ln P(v_{b_I}) - \sum_{i\mid b_i=b_I} \frac{P(c_i)}{P(v_{b_I})}\right)$

Hence the derivative is $\log\left(\frac{P(c_k)}{P(v_{a_k})}\right) - \log\left(\frac{P(c_I)}{P(v_{a_I})}\right)$. Similarly, for the situation where the clause type corresponding to p_k has the same verb as the last term, the derivative is $\log\left(\frac{P(c_k)}{P(n_{a_k})}\right) - \log\left(\frac{P(c_I)}{P(n_{a_I})}\right)$.

The derivative of the normalised mutual information is, as in Case Study 1, obtained using the quotient rule.

Using the above formulas, the variance of the normalised MI for the *Zuǒ Zhuàn* and *Gǔliáng Zhuàn* were estimated at 0.0002415962 and 0.001188776 respectively. Because the two samples are independent, to obtain the variance of the difference between the two, we simply sum up the two variances, resulting in an estimate of 0.001430372. The resulting confidence interval is (-0.3829, -0.2347). This suggests that the choice of negator has become more predictable between the two texts, and even at the lower end of the CI, the difference seems substantial at almost 0.24.

Confidence interval for the pointwise conditional entropy. The pointwise conditional entropy for the j-th negator is given as follows:

$$\begin{aligned} H_{v\mid n=n_j}(\theta) &= -\sum_{i\mid n_{a_i}=n_j} \frac{P(n=n_{a_i}, v=v_{b_i})}{P(n=n_{a_i})} \log\left(\frac{P(n=n_{a_i}, v=v_{b_i})}{P(n=n_{a_i})}\right) \\ &= -\sum_{i\mid n_{a_i}=n_j} \frac{P(c=c_i)}{P(n=n_{a_i})} \log\left(\frac{P(c=c_i)}{P(n=n_{a_i})}\right) \end{aligned} \tag{13}$$

Notice that the derivative of its summands with respect to $\frac{P(c=c_i)}{P(n=n_{a_i})}$ is $-\frac{1}{\ln 2}\left[\ln\left(\frac{P(c=c_i)}{P(n=n_{a_i})}\right) + 1\right]$. To get the derivative of the entire expression, by the chain rule, we need to find the derivative of $\frac{P(c=c_i)}{P(n=n_{a_i})} = P(c_i \mid n_{a_i})$, which differs across different conditions.

We first consider the case where $j = 1$, i.e. we derive the conditional entropy of verbs given that the negator is *bù*. The table below shows the derivative of $P(c_i|n_{a_i}) = P(c_i|n_1)$ with respect to p_k in various cases:

Case	Derivative of $P(c_i|n_{a_i})$ with respect to p_k
$v_{b_k} = v_{b_i}, n_{a_k} = n_1$	$\frac{P(n_1) - P(c_i)}{P(n_1)^2}$
$v_{b_k} \neq v_{b_i}, n_{a_k} = n_1$	$-\frac{P(c_i)}{P(n_1)^2}$
$n_{a_k} = n_2$	0

Thus the derivative of the entropy of verbs conditional on the negator being *bù* is $-\frac{1}{\ln 2} \sum_{i|n_{a_i}=n_1}$ $\left[\ln\left(\frac{P(c_i)}{P(n_1)}\right) + 1\right]\left[-\frac{P(c_i)}{P(n_1)^2} + \frac{I(v_{b_k} = v_{b_i})}{P(n_1)}\right]$.

For the derivative of the entropy of verbs conditional on the negator being *fú*, the situation is somewhat more complicated:

Case	Derivative of $P(c_i|n_2)$ with respect to p_k
$v_{b_i} \neq v_{b_I}, n_{a_k} = n_1$	$\frac{P(c_i)}{P(n_2)^2}$
$v_{b_i} = v_{b_I}, n_{a_k} = n_1$	$-\frac{P(n_2) - P(c_i)}{P(n_2)^2}$
$v_{b_k} = v_{b_i} \neq v_{b_I}, n_{a_k} = n_2$	0
$v_{b_k} \neq v_{b_i} \neq v_{b_I}, n_{a_k} = n_2$	$\frac{1}{P(n_2)}$
$v_{b_k} \neq v_{b_i} = v_{b_I}, n_{a_k} = n_2$	$-\frac{1}{P(n_2)}$

From these formulas, the estimates of the covariance matrix of for the *Zuǒ Zhuàn* are as follows:

$$\hat{Var}((H_{v|n=n_1}(\hat{\theta}), H_{v|n=n_2}(\hat{\theta}))^T) = \begin{bmatrix} 0.001596 & -3.532 \times 10^{-17} \\ -3.406 \times 10^{-18} & 0.00947 \end{bmatrix}$$

Similarly, the quantities for the *Gǔliáng Zhuàn* are as follows:

$$\hat{Var}((H_{v|n=n_1}(\hat{\theta}), H_{v|n=n_2}(\hat{\theta}))^T) = \begin{bmatrix} 0.00526 & -1.508 \times 10^{-16} \\ 1.427 \times 10^{-17} & 0.05996 \end{bmatrix}$$

Taking $C = (1, -1)^T$, the variance of the differences $H_{v|n=n_1}(\hat{\theta}) - H_{v|n=n_2}(\hat{\theta})$ is 0.00947 and 0.06521973 respectively for the two books. Simultaneous 95% CIs are (1.972, 2.444) and (2.71, 3.854) for the two differences respectively. The difference between the two differences has an estimated variance of 0.07628628 and a 95% CI of (-1.615, -0.5324). All the CIs exclude zero, so the answers to our research questions are affirmative.

5.2 Simulations

Simulations under ideal conditions. We compute 1000 simulations from the IID categorical model using the estimated probability estimates as the 'true' values, as in Case Study 1. The mean simulated difference between normalised MIs is -0.3084565, quite close to the 'true' mean, and the variance is 0.001548701 - quite close to our estimate. The Shapiro-Wilk test does not reject normality ($p = 0.1384$). The empirical coverage of the normal CIs is acceptable at 91.5%.

For the conditional entropies, the mean difference between the conditional entropy of verb on *bù* and *fú* are 2.274312 for the *Zuǒ Zhuàn* and 3.350070 for the *Gǔliáng Zhuàn*. The estimated variances are 0.01224777 and 0.06798214, and Shapiro-Wilk rejects normality only in the second case (p = 0.4002, 0.0001833), though a Q-Q plot (Supplementary Materials 5) shows the deviation is not serious. The empirical confidence level is 85.8% overall and 91.8% and 93.5% for the two books individually.

For the difference between the differences, the mean is -1.075759, quite close to the 'true' value of -1.073737. The empirical sample variance for the difference between the texts is 0.08026133 - only slightly higher than our estimate, the Shapiro-Wilk test again rejects normality ($p = 0.002225$) but a Q-Q plot again shows the deviation is mild, and the empirical confidence level is 91%.

Simulations under clustered conditions. When examining the raw data, one may notice that *fú*, the less frequent negator, sometimes appears many times in a document. For example, in the fifth entry in the tenth year of Duke Zhuāng, *fú* appeared four times and *bù* only once. This may be because of priming, similar semantics between sentences (all denote volitional actions), or some other reason. It seems reasonable to suppose that the distribution of negator-verb combinations varies across paragraphs. This violates the independence assumption, as data within a document are dependent.

To examine whether this violation adversely affects our estimates, we simulate 1000 datasets by resampling the documents with replacement: if there are D documents, then in each simulated dataset, we choose D of the documents with equal probability, allowing each document to appear more than once.

The mean simulated difference between normalised MIs is -0.3191, the variance is 0.001778684, and the CI's empirical coverage is 91.5%, so the clustering property does not affect the MI much. For the conditional entropies, the method performs somewhat worse. The mean difference between the conditional entropy of verb on *bù* and *fú* are 2.281 for the *Zuǒ Zhuàn* and 3.405 for the *Gǔliáng Zhuàn*. The estimated variances are 0.0128063896 and 0.114090982. The empirical confidence level is 77.1% overall and 89.6% and 85.9% for the two books individually. For the 'second-order' difference, the mean is -1.125, the empirical variance is 0.1283355, somewhat larger than our estimated value, and the empirical confidence level is relatively low at 81.7%. Apart from the first difference in conditional entropies, all of the above simulations are significantly nonnormal (p = 0.002411, 0.775375, 0.0002230431, 0.001496423) though again Q-Q plots show this is not serious.

As our estimated mean is farther away from zero than the 'true' value, along with the underestimated variance, the CI may be biased in our favour. Again, an alternative is to use the cluster-resampled data to derive a bootstrap confidence interval (Field & Welsh, 2007).

6 Conclusion

As an empirical discipline, linguistics aims at making the best generalisations on the basis of empirical data. But as in most disciplines, our data is only a sample of the (finite or infinite) data out there, and is bound by the constraints of our sample size and data collection methods. Hence, any quantity we calculate with this partial data should be carefully scrutinised, and we should quantify our uncertainty about the estimate as best we can. Moreover, rather than blindly following established methods for this quantification, we should ensure that the method is appropriate for the situation; that is, we should check whether our situation satisfies the method's assumptions and if not, whether the method is *robust* to the violations.

In this paper, we introduce a method of deriving confidence intervals for maximum likelihood estimates of information-theoretic quantities, and claim for its use whenever estimates of the covariance matrix of the model parameters are available and the sample size is large. In three case studies, we find that sometimes the conclusions we draw without uncertainty estimation still seem largely valid (as in Case Study 1 and 3), but other times, with smaller and noisier samples, they may turn out to be unsupported (as in Case Study 2).

We also suggest methods to test the method's validity, including whether we have sufficient sample size for the large-sample approximation to be appropriate, and whether the incorrect assumptions of the underlying model undermine the validity of the confidence interval estimate. In Case Study 1, we find that the method is largely valid and robust to assumption violations, and the one place where it underperforms does not change our conclusions substantially. In Case Study 2, the simulations reveal that the amount of noise in the data is even greater than our method reveals. In Case Study 3, we find that the method is sufficient under model assumptions, but once we take into account dependencies in the data, the method partially underperforms. Other methods, such as bootstrap methods, may be useful when we find our normal confidence interval insufficient.

We have demonstrated how the method may be used with a wide variety of ITMs, including marginal entropy, functional load, KLD and mutual information. Linguists working with similar data may take the formulas we derive in this paper; if not, the general method can be done either by deriving the formulas by hand, or with numerical approximation methods like the R package `numDeriv` (Supplementary Materials 6).

References

Albright, A., & Do, Y. (2013). *Three biases for learning phonological alternations.* Paper presented at the Twenty-First Manchester Phonology Meeting, Manchester.

Bates, D., Maechler, M., Bolker, B., Walker, S., et al. (2015). Fitting linear mixed-effects models using lme4. *Journal of Statistical Software*, *67*(1), 1–48.

Bauer, R. S., & Benedict, P. K. (1997). *Modern Cantonese phonology* (Vol. 102). Walter de Gruyter.

Davison, A. C., & Hinkley, D. V. (1997). *Bootstrap methods and their application* (Vol. 1). Cambridge University Press.

Denwood, P. (1999). *Tibetan* (Vol. 3). John Benjamins Publishing.

Do, Y., & Lai, R. K. Y. (2019). Measuring phonological distance in a tonal language: An experimental and computational study with Cantonese. *Proceedings of the Society for Computation in Linguistics*, *2*(1), 371–372.

Efron, B., & Tibshirani, R. J. (1994). *An introduction to the bootstrap.* CRC press.

Esukhia Development Team. (2019, June). *pyewts v.0.1.1.* Retrieved from `https://pypi.org/project/pyewts/`

Field, C. A., & Welsh, A. H. (2007). Bootstrapping clustered data. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *69*(3), 369–390.

Garson, N., & Germano, D. (2004, January). *Extended Wylie transliteration scheme.* Tibetan and Himalayan Digital Library. Retrieved from `http://www.thlib.org/reference/transliteration/#!essay=/thl/ewts/` doi: 10.5281/zenodo.803268

Germano, D., Garrett, E., & Weinberger, S. (2017, June). *UVA Tibetan spoken corpus.* Retrieved from `https://doi.org/10.5281/zenodo.803268` doi: 10.5281/zenodo.803268

Gilbert, P., Gilbert, M. P., & Varadhan, R. (2006). *The numDeriv package.*

Gu, B. (1998). *Xin yi Guliang Zhuan [a new translation of the guliang zhuan].* Sanmin Shuju Yinhang.

Hogg, R. V., McKean, J., & Craig, A. T. (2005). *Introduction to mathematical statistics.* Pearson Education.

Lee, J. L., Chen, L., & Tsui, T.-H. (2016). PyCantonese: Developing computational tools for Cantonese linguistics.

Lee, S. M. S., & Young, G. A. (1995). Asymptotic iterated bootstrap confidence intervals. *The Annals of Statistics*, 1301–1330.

Liu, L. (2004). *Xianqin fouding fuci 'bu', 'fu' zhi bijiao [a comparison of the 'bu' and 'fu' negating adverbs in the pre-Qin era]* (Unpublished master's thesis). Shaanxi Normal University.

Luke, K. K., & Wong, M. L. (2015). The Hong Kong Cantonese corpus: design and uses. *Journal of Chinese Linguistics*, *25*(2015), 309–330.

Pulleyblank, E. G. (2010). *Outline of classical Chinese grammar.* Vancouver: UBC Press.

Rao, C. R. (1973). *Linear statistical inference and its applications* (Vol. 2). Wiley New York.

Tournadre, N., & Dorje, S. (2003). *Manuel de tibétain standard.* L'Asiathèque-Maison des langues du monde.

Wallis, S. (2013). Binomial confidence intervals and contingency tests: mathematical fundamentals and the evaluation of alternative methods. *Journal of Quantitative Linguistics*, *20*(3), 178–208.

7 Supplementary materials

7.1 Multivariable calculus

Partial derivatives. The partial derivative of a function $f(x_1,x_2,..,x_n)$ with respect to x_i, denoted by $\frac{\partial f(x_1,x_2,...,x_n)}{\partial x_i}$, which we do not formally define here, is the derivative of that function treating all of the variables other than x_i as constants.

The Hessian matrix. The Hessian matrix is of a scalar-valued function $y = f(x_1,x_2,...,x_n)$ is defined as follows:

$$\begin{bmatrix} \frac{\partial^2 y}{\partial x_1^2} & \frac{\partial^2 y}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 y}{\partial x_1 \partial x_n} \\ \frac{\partial^2 y}{\partial x_2 \partial x_1} & \frac{\partial^2 y}{\partial^2 x_2} & \cdots & \frac{\partial^2 y}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 y}{\partial x_n \partial x_1} & \frac{\partial^2 y}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 y}{\partial x_n^2} \end{bmatrix}$$

The Jacobian matrix. The Jacobian matrix is the multi-dimensional analogue of the first derivative. For a vector-valued function $(y_1,y_2,...y_m) = F(x_1,x_2,...,x_n)$, the Jacobian matrix is as follows:

$$DF = \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_1}{\partial x_2} & \cdots & \frac{\partial y_1}{\partial x_n} \\ \frac{\partial y_2}{\partial x_1} & \frac{\partial y_2}{\partial x_2} & \cdots & \frac{\partial y_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial y_m}{\partial x_1} & \frac{\partial y_m}{\partial x_2} & \cdots & \frac{\partial y_m}{\partial x_n} \end{bmatrix}$$

The multivariable chain rule. If F and G are vector-valued functions, then the derivative of their composition is the product of their derivatives, i.e. $D(F \circ G) = DF \times DG$.

7.2 Supplementary materials to Case Study 1

Q-Q plots for the ideal simulation. The relevant Q-Q plot is presented for the quantity for which the Shapiro-Wilk test rejects normality.

Results for the sequential dependence simulations. For the entropies, the average vector of differences is $(1.016, 1.224, 1.353, 0.2077, 0.3368, 0.1291)^T$. The empirical covariance matrix

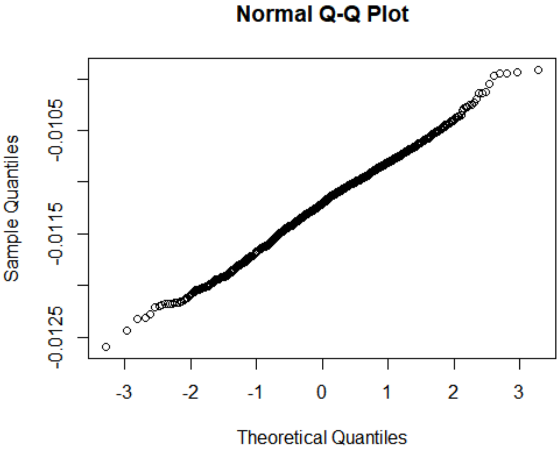


Figure 1: Q-Q plot for the functional load difference between nucleus and tone.

is as follows:

$$\begin{bmatrix} 1.227\times10^{-5} & 4.605\times10^{-6} & 5.872\times10^{-6} & -7.667\times10^{-6} & -6.4\times10^{-6} & 1.267\times10^{-6} \\ 4.605\times10^{-6} & 1.577\times10^{-5} & 5.267\times10^{-6} & 1.116\times10^{-5} & 6.629\times10^{-7} & -1.05\times10^{-5} \\ 5.872\times10^{-6} & 5.267\times10^{-6} & 6.722\times10^{-6} & -6.042\times10^{-7} & 8.508\times10^{-7} & 1.455\times10^{-6} \\ -7.667\times10^{-6} & 1.116\times10^{-5} & -6.042\times10^{-7} & 1.883\times10^{-5} & 7.063\times10^{-6} & -1.177\times10^{-5} \\ -6.4\times10^{-6} & 6.629\times10^{-7} & 8.508\times10^{-7} & 7.063\times10^{-6} & 7.251\times10^{-6} & 1.878\times10^{-7} \\ 1.267\times10^{-6} & -1.05\times10^{-5} & 1.455\times10^{-6} & -1.177\times10^{-5} & 1.878\times10^{-7} & 1.196\times10^{-5} \end{bmatrix}$$

Normality is not rejected ($p = 0.7939, 0.6214, 0.2255, 0.8751, 0.5421, 0.9407$). The empirical coverages were 0.996, 0.991, 0.992, 0.992, 0.995, 0.991 separately and 96.6% overall, so we can trust the results.

For the functional load, the average vector of differences is $(0.08584, 0.09571, 0.07462, 0.009877, -0.01122, -0.0211)^T$ and the covariance is matrix is as follows:

$$\begin{bmatrix} 2.331\times10^{-7} & 1.455\times10^{-7} & 1.301\times10^{-7} & -8.761\times10^{-8} & -1.03\times10^{-7} & -1.538\times10^{-8} \\ 1.455\times10^{-7} & 2.405\times10^{-7} & 1.271\times10^{-7} & 9.505\times10^{-8} & -1.84\times10^{-8} & -1.134\times10^{-7} \\ 1.301\times10^{-7} & 1.271\times10^{-7} & 2.264\times10^{-7} & -3.026\times10^{-9} & 9.635\times10^{-8} & 9.938\times10^{-8} \\ -8.761\times10^{-8} & 9.505\times10^{-8} & -3.026\times10^{-9} & 1.827\times10^{-7} & 8.458\times10^{-8} & -9.807\times10^{-8} \\ -1.03\times10^{-7} & -1.84\times10^{-8} & 9.635\times10^{-8} & 8.458\times10^{-8} & 1.993\times10^{-7} & 1.148\times10^{-7} \\ -1.538\times10^{-8} & -1.134\times10^{-7} & 9.938\times10^{-8} & -9.807\times10^{-8} & 1.148\times10^{-7} & 2.128\times10^{-7} \end{bmatrix}$$

There is very little difference with the version without sequential dependence, especially if we consider only the diagonal of the covariance matrix. The coverages are 1, 1, 1, 0.924, 0.817, 0.302 separately and 26.2% overall. Normality is not rejected ($p = 0.4431, 0.8799, 0.5048, 0.6183, 0.8155, 0.1769$).

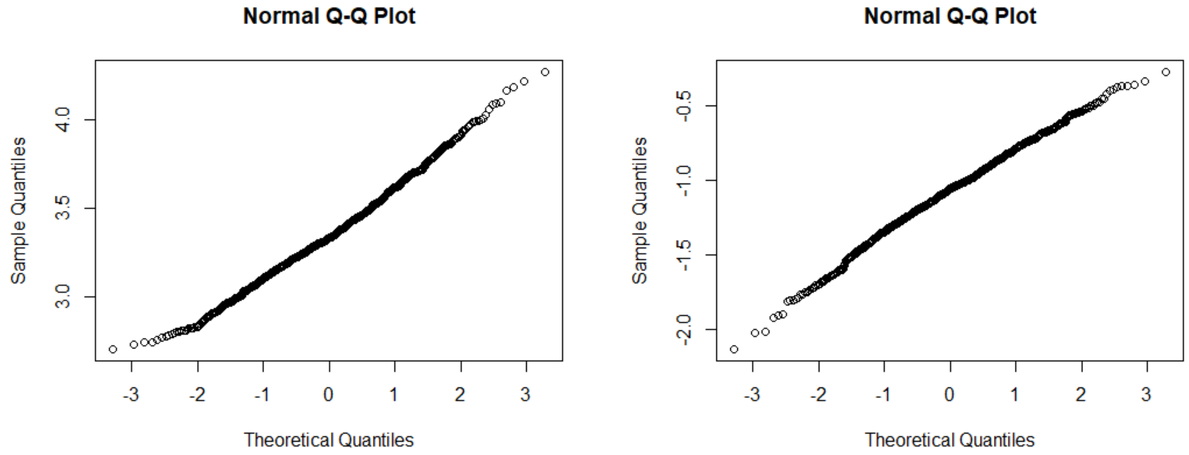


Figure 2: Q-Q plots for the *Gŭliáng Zhuàn*'s pointwise conditional information difference (left) and the difference between the two books (right) in the ideal simulation.

7.3 Supplementary materials in Case Study 3

Annotation scheme. Each sentence containing *bù* or *fú* were annotated to note what its verb was. We excluded the three lexicalised set phrases *bùrú* 不如 ('why not'), *búyì ... hū* 不亦 ... 乎 ('is (it) not also ...?') and 不然 *bùrán* ('not so') from consideration, although cases of *bùrú* with its original meaning as a negator + verb combination meaning 'not like, not to the standards of' were included. For example, the (a) sentence below is included in our sample but not the (b) sentence:

(1) a. 猶 不 如 人
yóu bù rú rén
still NEG like others
'(I) am still not as (strong as) others.' (*Zuǒ Zhuàn*, 30th year of Duke Xī, paragraph 8)

b. 不 如 殺 之
bù rú shā zhī
NEG like kill 3sg
'Why not kill him?' (*Zuǒ Zhuàn*, 30th year of Duke Zhuāng, paragraph 7)

Q-Q plots. The relevant Q-Q plots are presented for the quantities for which the Shapiro-Wilk test rejects normality. Most of them seem to display fairly mild violations.

7.4 Numerical evaluation in R

In practice, not all linguists may be able to calculate derivatives directly, as we have done in the previous simulations, especially when the formula is complex. Moreover, it is useful to be able to check our formulas even if we can derive them. In such situations, we may wish to calculate

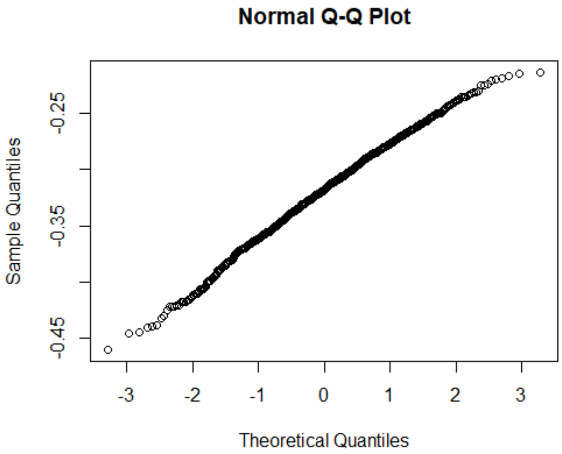
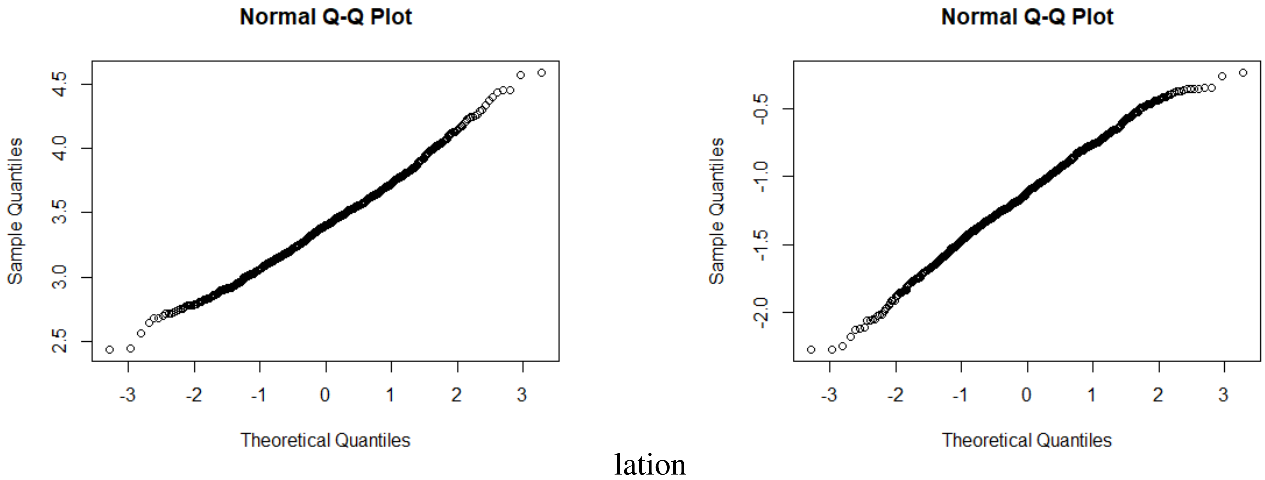


Figure 3: Q-Q plots for the normalised MI difference in the clustered simulation.



lation

Figure 4: Q-Q plots for the *Gŭliáng Zhuàn*’s pointwise conditional information (left) and the difference between the two (right) in the clustered simulation.

the required quantities numerically. Here, we demonstrate how this may be done in the Tibetan tone/aspiration example using the R package `numDeriv` (Gilbert, Gilbert, & Varadhan, 2006).

Numerical evaluation of the Hessian. We have given an explicit formula for the IID categorical distributions, which is probably the most common distribution used in practice. For probabilities derived from packages (such as mixed models), the Hessian or its inverse is usually included somewhere in the model object returned in R. Nevertheless, for completeness' sake, we demonstrate how Hessians can be calculated in R.

A function must first be created, the first argument of which is the parameter vector. In this case, the parameter vector `p_vec` consists of three components: the probabilities of high/aspirated, high/nonaspirated and low/aspirated. The second argument is the vector of frequencies: high/aspirated, high/nonaspirated, low/aspirated, low/nonaspirated. We then evaluate the Hessian using the `hessian` function of `numDeriv` and invert it using the `solve` function:

```
library(numDeriv)
neg_loglikelihood = function(p, freqs = freq_vec) -log(p[1])*
    freqs[1] - log(p[2])*freqs[2] - log(p[3])*freqs[3] - log(1 -
    p[1] - p[2] - p[3])*freqs[4]
var_hat_numeric = solve(hessian(neg_loglikelihood, p_vec))
```

The resulting matrix is very similar to the exact version we have computed in Section 2; the largest absolute error is 2.153×10^{-18}.

Numerical evaluation of the Jacobian. To evaluate the Jacobian matrix of the entropies of tone and aspiration, we first create a function for finding the entropy. Then, using this function, we create a function that takes the probability vector and returns the entropies of tone and aspiration as a vector. We then use the function `jacobian` to evaluate the Jacobian.

```
find_entropy = function(p) - p * log(p, 2) - (1 - p) * log(1 - p
    , 2)
toneasp_entropies_func = function(p, freqs = freq_vec) c(find_
    entropy(p[1] + p[2]), find_entropy(p[1] + p[3]))
entropies_jacobian_numeric = jacobian(toneasp_entropies_func, p_
    vec)
```

The numerical approximation is still quite accurate: The largest difference with the true value is only 6.683×10^{-11}.