Using Approximate Bayesian Computation for estimating parameters in the cue-based retrieval model of sentence processing

Shravan Vasishth

Department of Linguistics, University of Potsdam, Potsdam, Germany

July 15, 2019

Author Note

Correspondence: vasishth@uni-potsdam.de.

Abstract

In this methods paper, we explain how prior and posterior predictive distributions of reading times are generated from the cue-based retrieval of Lewis & Vasishth, 2005. Prior predictive distributions of reading time are generated from the model by defining a mildly informative prior on the parameter of interest (here, the latency factor). The posterior predictive distribution involves two steps: first, Approximate Bayesian Computation (ABC) is used with rejection sampling to compute the posterior distribution of the parameter of interest. This posterior distribution is then used to generated a posterior predictive distribution of reading times and of the effects predicted by the model. The ABC method of parameter estimation is superior to conventionally used approaches such as grid search, because model predictions take into account the uncertainty of the parameter value.

Keywords: Approximate Bayesian Computation; Bayesian parameter estimation; prior and posterior predictive checks; cue-based retrieval; sentence processing

Using Approximate Bayesian Computation for estimating parameters in the cue-based retrieval model of sentence processing

Introduction

This paper explains the method used in Jäger, Mertzen, Van Dyke, and Vasishth (2019) to estimate the latency factor parameter in the cue-based retrieval model of Engelmann, Jäger, and Vasishth (2019), when evaluating the model's predictions to the observed data from Dillon, Mishler, Sloggett, and Phillips (2013) and our larger-sample replication attempt (Jäger et al., 2019). The source code and data associated with the methods reported here and the paper by Jäger et al. (2019) are available from https://osf.io/reavs/.

The cue-based retrieval model of Engelmann, Jäger, and Vasishth, 2019

This model is a simplified version of the Lisp-based model described in Lewis and Vasishth (2005). This simplified version is written in R and abstracts away from the individual incremental parsing steps of the original model, and focuses instead only on the retrieval time and retrieval accuracy computations, given some retrieval cues and candidate chunks in memory that could match the retrieval cues.

Table 1 shows the parameter values used in the recent large-sample evaluation (approximately 100 published reading experiments) of the cue-based retrieval model described in Engelmann et al. (2019). Here, we follow the practice that was adopted in Lewis and Vasishth (2005), of holding all the parameters constant to their default value. The only exception is the latency factor parameter, which scales retrieval time to the millisecond reading time scale. The reason for holding the parameters constant is to avoid overfitting to the particular data being considered.

Bayesian parameter estimation

Here, we provide some of the background needed to understand the parameter estimation approach described below. In the Bayesian modeling framework, given a vector of

Table 1
Model parameters, their default values, and the values used in the simulation of the studies discussed in Engelmann et al., 2019.

Parameter	Name	Default	Simulation
F	latency factor	0.2	$[0.1, 0.25]$
f	latency exponent	1	1
τ	retrieval threshold	-1.5	-1.5
d	decay rate	0.5	0.5
ANS	activation noise	0.2	0.2
MAS	maximum associative strength	1	1.5
MP	mismatch penalty	1	0.25
β	base-level activation	0	0

data y and a vector of model parameters θ that have prior distributions $p(\theta)$ defined on them, a likelihood function for the data $p(y \mid \theta)$ and the priors allow us to compute the posterior distribution of the parameters given the data, $p(\theta \mid y)$. This is possible because of Bayes' rule, which states that the posterior is proportional to the likelihood times the prior:

$$p(\theta \mid y) \propto p(y \mid \theta)p(\theta) \tag{1}$$

The posterior distributions of parameters are generally computed using Monte Carlo Markov Chain methods. Examples are Gibbs sampling, Metropolis-Hastings, and (more recently) Hamiltonian Monte Carlo.

The likelihood and the priors together constitute the model, which we will call $\mathcal{M}$ hereafter. Given a particular model $\mathcal{M}$, one important question is: what predictions does the model make? The model makes two kinds of predictions: a priori predictions, before any data have been taken into account; and a posteriori predictions, after the data have been taken into account. The distributions of these two kinds of predictions are called *prior predictive distributions*, and *posterior predictive distributions*, respectively.

The prior predictive distribution can be computed by drawing random samples of the parameters $\tilde{\theta}$ from $p(\theta)$, and then using these values to simulate data $\tilde{y}$ from the likelihood

$p(y \mid \tilde{\theta})$.

The posterior predictive distribution $p(y_{pred} \mid y)$ can be computed once we have the posterior distribution of the parameters, $p(\theta \mid y)$. Here, we assume that past and future observations are conditionally independent given θ.

$$p(y_{pred} \mid y) = \int p(y_{pred} \mid \theta) p(\theta \mid y) \, d\theta \tag{2}$$

An important point to note here is that we are conditioning y_{pred} only on y. We do not condition on the unknown parameters θ; we simply integrate these unknown parameters out. This allows us to take the uncertainty of the posterior distributions of the parameters into account, giving us more realistic estimates of the predictions from the model. Contrast this with a situation where we condition on, e.g., maximum likelihood estimates of the parameters; that is, we condition on a point value, not taking the uncertainty of that estimate into account.

Approximate Bayesian Computation

Approximate Bayesian Computation (ABC) is a method for estimating posterior distributions of parameters in a model. ABC is useful when Bayes' rule cannot be employed to draw samples from the posterior distributions; this situation arises when the generative model cannot be easily expressed as a likelihood function. For extensive treatments of the theory and practical aspects of ABC, see Sisson, Fan, and Beaumont (2018), Palestro, Sederberg, Osth, Van Zandt, and Turner (2018). The algorithm used here is rejection sampling; see Listing 1 for pseudo-code describing the algorithm.

Algorithm 1: ABC using rejection sampling. Shown is the case where we need to sample posterior values for a single parameter θ. Each iteration of the algorithm consists of drawing a single random sample from a prior distribution for the parameter (here, $Beta(2, 6)$), and then generating the predicted mean effect from the model using that sampled parameter value. If the predicted mean effect is near the observed data (in our implementation, if the predicted effect lies within one standard error of the mean effect of interest), then accept the sampled parameter value; otherwise reject that sampled value. This process is repeated until we have sufficient samples from the posterior distribution of the parameter. These samples therefore constitute the posterior distribution of the parameter.

```
Input: Tolerance bounds lower and upper from data
begin
    for i in 1:N_Simulations do
        Take one sample from prior $\pi(\theta)$;
        Generate predicted mean effect $\tilde{\tilde{y}} \sim Model(\theta)$;
        if $lower \leq \tilde{\tilde{y}} \leq upper$ then
            Save $\theta$ value as sample from posterior;
        end
        else
            Discard $\theta$ sample;
        end
    end
end
```

Bayesian estimates of the latency factors

Step 1: Define a prior for the parameter

We begin by defining a prior distribution on the latency factor in the cue-based retrieval model. Several priors can be considered: a Uniform prior or a Beta prior are examples. For illustration, we use the Beta(2,6) prior. As shown in Figure 1, this is a relatively uninformative prior which downweights very small and very large values of the latency factor parameter.

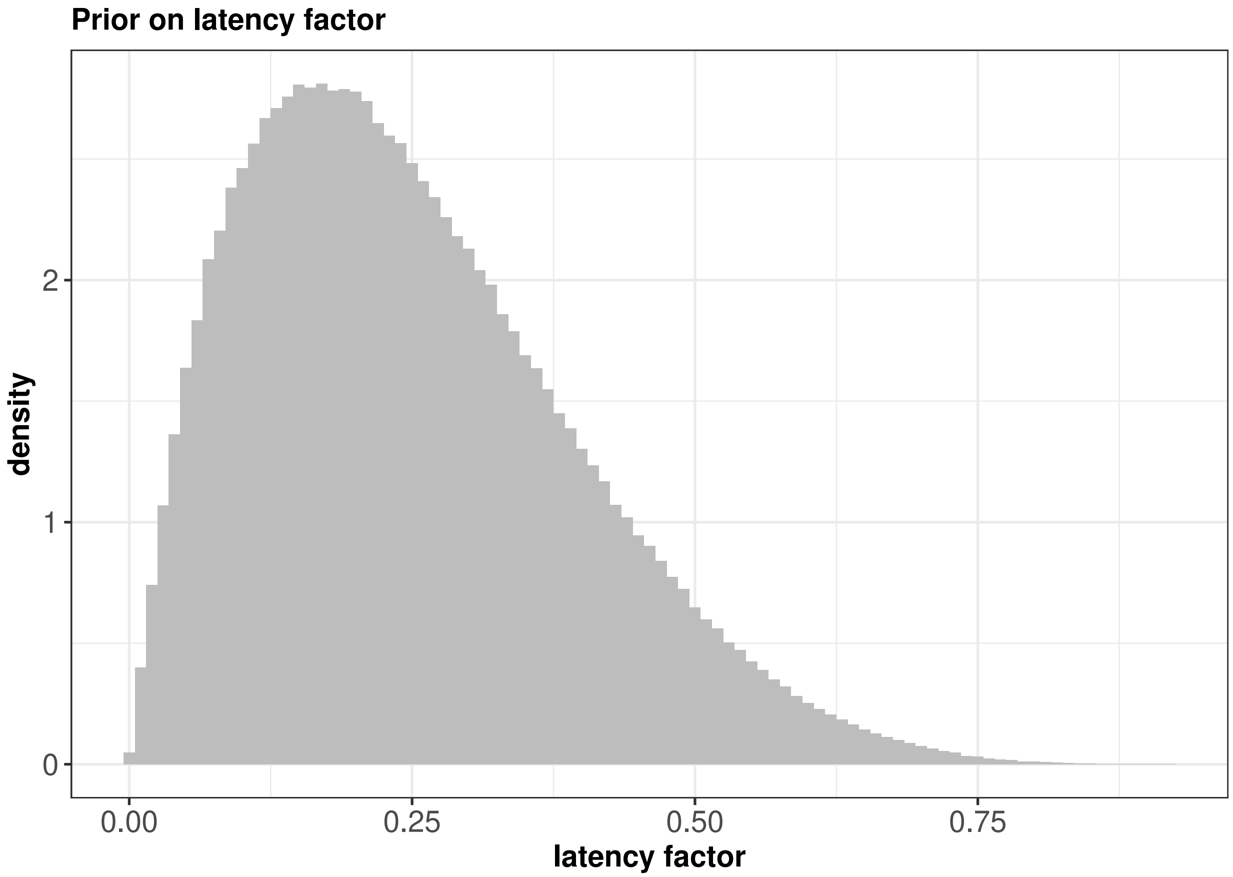

Figure 1. A Beta(2,6) prior on the latency factor.

The estimates from data for ungrammatical conditions. In the ungrammatical conditions of the Dillon et al. (2013) data, the estimate of the interference effect in agreement conditions is -60 ms, Credible interval (CrI) [-112, -5] ms. Taking a normal approximation, this implies an effect coming from the distribution $Normal(-60, 33^2)$.

Similarly, the estimate of the interference effect in reflexive conditions is -18 ms, CrI [-72, 36] ms, which corresponds approximately to the $Normal(-18, 27^2)$.

We can use these normal approximations to define a lower and upper bound for the ABC algorithm: one standard deviation about the observed mean. The acceptance criterion of the ABC algorithm is that the predicted value generated by the model lies within one standard deviation of the sample mean from the data.

In the Jäger et al. (2019) data, the estimate of the interference effect in agreement conditions is -22 [-46, 3], which can be approximated by the $Normal(-22, 13^2)$. The estimate in reflexive conditions is -23 [-48, 2], which can be approximated as the $Normal(-23, 13^2)$.

Step 2: Compute posterior distributions of the latency factor using ABC rejection sampling

Figure 2 shows the posterior distributions of the latency factor parameter for ungrammatical agreement and reflexive conditions in Dillon et al. (2013) and Jäger et al. (2019). The estimates for the Dillon et al. (2013) data-set have wider uncertainty than those for Jäger et al. (2019) because the uncertainty of the facilitatory interference effects in the data is relatively large.

Step 3: Generate posterior predicted data

Having estimated the posterior distributions of the latency factor for the two data-sets in the two conditions (agreement and reflexives), we can now generate posterior predicted data from the model. We use the posterior distributions of the latency factor to generate the posterior predictive distribution of the interference effect in these experimental conditions. These posterior predictive distributions are shown in Figure 3.

The ABC method can be generalized using other, more efficent sampling approaches (e.g., Metropolis-Hastings) to sample the posterior from more than one parameter. The method is computationally expensive but the advantages afforded by taking parameter uncertainty into account in the predictions is very valuable.

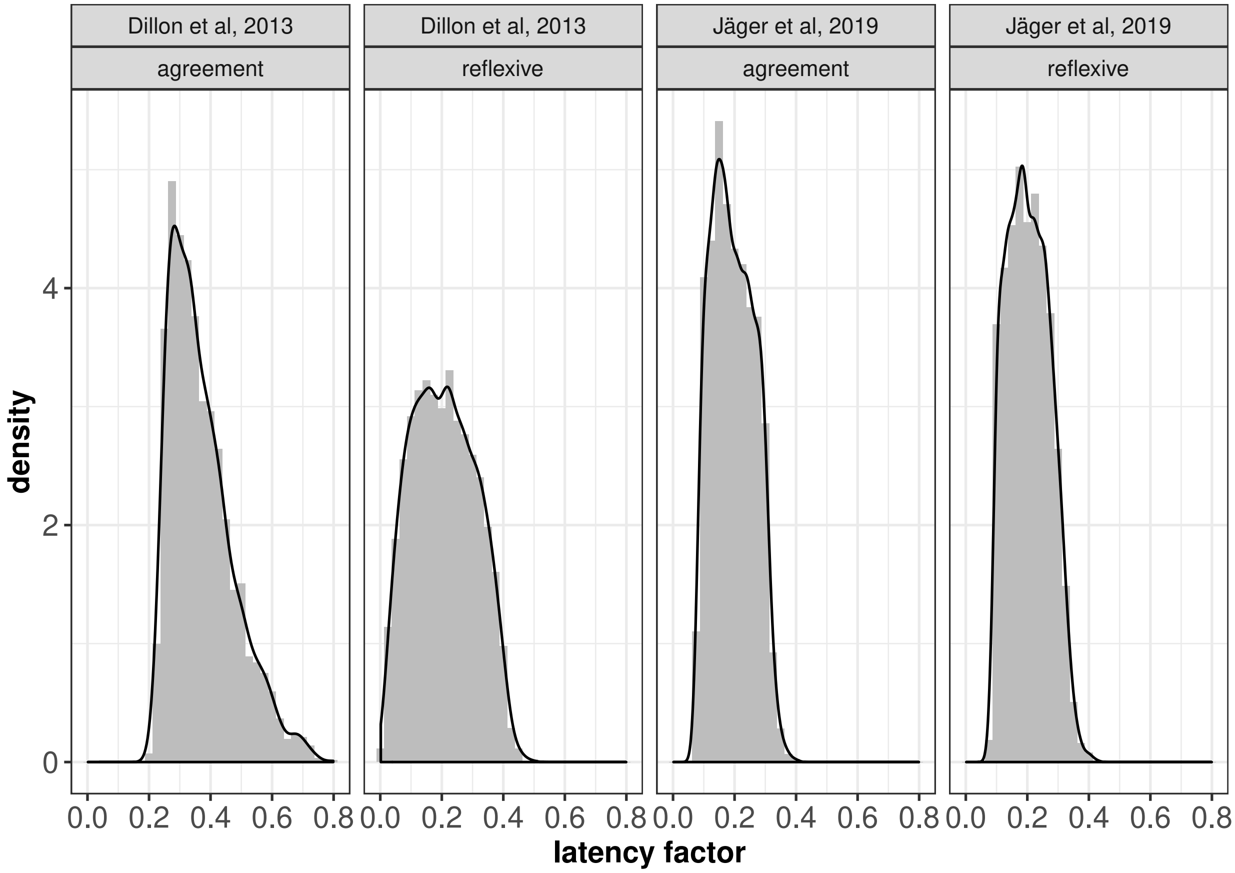

Figure 2. The posterior distributions of the latency factor parameters for agreement and reflexive conditions using the original Dillon et al., 2013 data (40 participants, 48 items) and our own Jäger et al., 2019 replication data (181 participants, 48 items).

Conclusion

In closing, the ABC method is a powerful tool for parameter estimation in models like the cue-based retrieval model, which cannot be easily expressed as a likelihood. As discussed in (Kangasrääsiö, Jokinen, Oulasvirta, Howes, & Kaski, 2019), this approach should be adopted more widely in psycholinguistics and related areas because it allows us to take parameter uncertainty into account when evaluating model predictions. This will yield more realistic predictions than using point values for parameters.

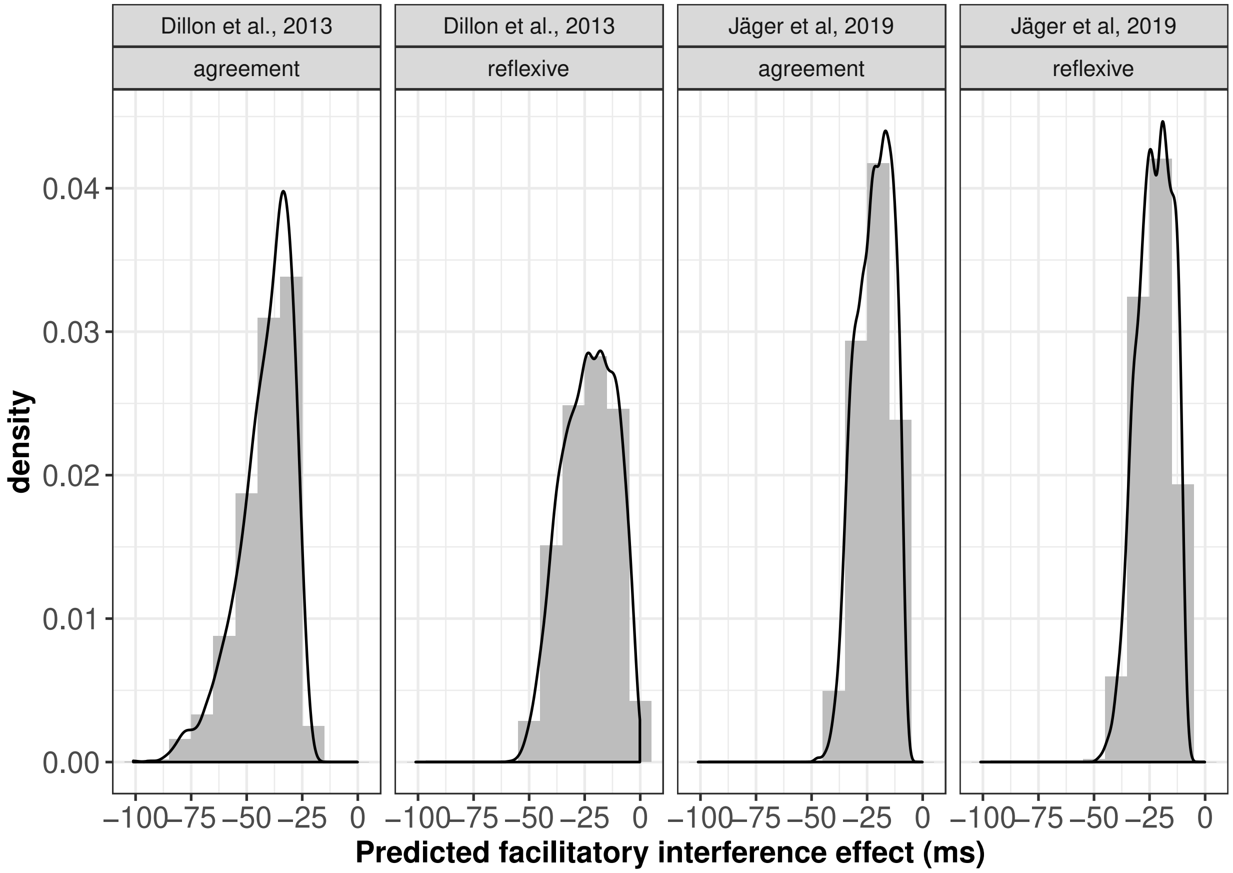

Figure 3. The posterior predictive distributions of the facilitatory interference in ungrammatical agreement and reflexive conditions, derived using the posterior distributions of the latency factor parameter.

Acknowledgements

The research reported here was partly funded by the Volkswagen Foundation through grant 89 953; and the Deutsche Forschungsgemeinschaft (German Science Foundation), Collaborative Research Center - SFB 1287, project number 317633480 (*Limits of Variability in Language*) through projects B3 (PIs: Ralf Engbert and Shravan Vasishth) and Q (PIs: Shravan Vasishth and Ralf Engbert). Thanks go to Garrett Smith for discussions. Thanks also to Brian Dillon for his review of the paper accompanying this Methods paper; the present Methods paper addresses his concerns about parameter estimation.

References

Jäger, L. A., Mertzen, D., Van Dyke, J. A., & Vasishth, S. (2019). *Interference patterns in subject-verb agreement and reflexives revisited: A large-sample study.* Accepted pending minor revisions.

Engelmann, F., Jäger, L. A., & Vasishth, S. (2019). The effect of prominence and cue association in retrieval processes: A computational account. *Cognitive Science.* Accepted pending minor edits.

Dillon, B. W., Mishler, A., Sloggett, S., & Phillips, C. (2013). Contrasting intrusion profiles for agreement and anaphora: Experimental and modeling evidence. *Journal of Memory and Language*, *69*, 85–103.

Lewis, R. L., & Vasishth, S. (2005). An activation-based model of sentence processing as skilled memory retrieval. *Cognitive Science*, *29*(3), 375–419.

Sisson, S. A., Fan, Y., & Beaumont, M. (2018). *Handbook of approximate Bayesian computation.* Chapman and Hall/CRC.

Palestro, J. J., Sederberg, P. B., Osth, A. F., Van Zandt, T., & Turner, B. M. (2018). *Likelihood-free methods for cognitive science.* Springer.

Kangasrääsiö, A., Jokinen, J. P., Oulasvirta, A., Howes, A., & Kaski, S. (2019). Parameter inference for computational cognitive models with Approximate Bayesian Computation. *Cognitive Science*, *43*(6), e12738.