Course project

Bayesian statistical inference II

16.07.2023

Agreement attraction in sentence processing: Grammaticality-judgment study

In a grammaticality-judgment study, the participants are asked to make a response “grammatical” or “ungrammatical” after they have read a sentence. For example, the participant sees a sentence *The key to the cabinets are rusty* word-by-word, when they reach the final word *rusty*, they have to press either “F” or “J” as quickly as possible; where “F” represents “Grammatical” and “J” represents “Ungrammatical”. Hence, two variables are recorded in the task: the response (either **Grammatical** or **Ungrammatical**), and the response time (time taken to press one of the keys after the onset of the final word of the sentence).

This grammaticality-judgment paradigm is used to study agreement attraction phenomenon in sentence processing.

Consider the following pair of ungrammatical sentences:

(a) Attractor condition

*The key to the cabinets are rusty.

(b) Baseline condition

*The key to the cabinet are rusty.

A typical observation is the **attraction effect**: the response times in the attractor condition (a) are slower than in the baseline conditions. One possible reason is that the in condition (a), the evidence for ungrammaticality is equivocal. Because the noun “the key” suggests that the verb should be singular and “the cabinets” indicate that the verb should be plural. In contrast, both the nouns in condition (b) indicate in the same direction that the verb should be singular, not plural.

Suppose we conducted a new experiment and collected the grammaticality-judgment data. You can load the dataset from “project-data.Rda”.

We will try different models to infer about the underlying process that has generated the above data.

Problem 1: Implement a simple accumulator model

You can model the above data using a lognormal race model. The race model would assume that the grammaticality-judgment data is generated by the race of two accumulators, namely 'grammatical' and 'ungrammatical' accumulators.

A testable hypothesis is the following:

For the ungrammatical accumulator, the speed of evidence accumulation should be slower for the **attractor** condition because the evidence from number marking on nouns is unequivocal in this condition.

1. Fit the lognormal race model (see 20.1.3 in the book for reference) under the assumpion the (mean) drift rate of the ungrammatical accumulator is slower in the **attractor** condition than the the **baseline** condition.
2. Graph the parameter estimates. What can you infer from the estimates of the drift rate parameters?
3. Compare the above model against an 'alternative' lognormal race model. This alternative model assumes that the (mean) drift rate of the ungrammaical accumulator is faster in the **attractor** condition compared to the **baseline** condition.

Problem 2: Implement a feature migration model

To successfully comprehend a sentence, the reader needs to maintain certain words in memory. The words stored in memory are used for dependency completion: the process of linking the words that are syntactically related to each other.

For example, to comprehend the sentence "the key to the canibets was rusty", the reader needs to maintain the phrase"the key" in memory until they encounter the verb "was", otherwise it would be difficult to figure out "what was on the table?".

But when words/chunks are stored in memory, their representation gets degraded with time due to limited working-memory capacity in humans. The feature migration model formulates how this representation degradation takes place.

According to the feature migration theory, the plural feature of "the cabinets" in condition (a) percolates to the subject noun "the key" with some probability θ. No such feature migration happens in condition (b).

suppose there were total N trails in an experiment. In $\theta \times N$ trials, the plural feature of "the cabinet" would migrate to the subject, making the subject noun plural. The plural subject in these trials creates an illusion of grammaticality and consequently causes incorrect and slower judgments about the grammaticality of the sentence. However, in the remaining $(1 - \theta) \times N$ trials, the judgment times are same as in condition (b).

The overall effect is that the response times are slower in the attraction condition compared to the baseline condition.

The model implements this feature migration assumption using a mixture of lognormals.

The response time in trail i in condition (a) is assumed to come from a mixture of two lognormal distributions:

$$RT_i \sim \begin{cases} Lognormal(\mu+\delta,\sigma), & \text{if } z_i = 1 \\ Lognormal(\mu,\sigma), & \text{if } z_i = 0 \end{cases} \text{ where } z_i \sim Bernoulli(\theta) \tag{1}$$

Similarly,

$$Response_i \sim \begin{cases} Bernoulli(P_1), & \text{if } z_i = 1 \\ Bernoulli(P_2), & \text{if } z_i = 0 \end{cases} \text{ where } z_i \sim Bernoulli(\theta) \tag{2}$$

where P_1 and P_2 represent the probability of responding "Grammatical".

The model is constrained as follows:

$\delta > 0$

$P_1 > P_2$

In condition (b):

$$RT_i \sim Lognormal(\mu,\sigma) \tag{3}$$

and,

$$Response_i \sim Bernoulli(P_2) \tag{4}$$

1. Choose reasonable priors on P_1, P_2, μ, δ, σ, and θ.
2. Generate prior predictions from the model. Does the prior predictive data capture some basic properties of the observed data?
3. Fit the mixture model and graph the posterior distribution for each parameter.
4. Compare the above model against a 'null' mixture model that assumes that the rate of feature migration θ is equal to zero.

Problem 3: Model comparison

Evaluate the performance of the lognormal race model and the feature migration model. You can use either Bayes factors or some kind of cross-validation method to compare their performance. What can you infer from the model evaluation?

Problem 4: Multinomial processing tree

Can you come up with a new multinomial processing tree to model the given data? You have total freedom in choosing your assumptions.

State your assumptions clearly. Show the prior predictions of your model.

Fit the model and compare it against the lognormal race model and the mixture model above.