9.19: Computational Psycholinguistics, Pset 5
due 22 November 2023

8 November 2023

1 Logistic Regression

The Colab notebook for this problem is available at https://colab.research.google.com/drive/1DvtMo9PpbBdD9s5StcVafWm6hmAmdDZ5?usp=sharing

This problem is a continuation of our look at the English dative alternation. You will work with a dataset of annotated examples of the dative alternation from spontaneous speech in conversation, reported in Bresnan et al. (2007) and made available through the `languageR` package (Baayen, 2007). The Colab notebook for this problem automatically downloads this dataset as `dative.csv`.

As you've seen before, the dative alternation involves ditransitive verbs, and

(1) a. Kim mailed $\overbrace{\text{me}}^{\text{Recipient}}$ $\overbrace{\text{a gift}}^{\text{Theme}}$. [D(ouble) O(bject)]
b. Kim mailed $\underbrace{\text{a gift}}_{\text{Theme}}$ to $\underbrace{\text{me}}_{\text{Recipient}}$. [P(repositional) D(ative)]

The theme and recipient arguments are, respectively, what gets acted upon (usually transferred in physical location or possession), and the recipient or destination of the action. One qualitative intuition often reported about the dative alternation is that cases where the recipient argument is a large phrase (as measured, e.g., in number of words) are awkward, such as the below example:

(2) ?Kim mailed everyone who had attended the party yesterday a gift.

Alternatively, some researchers have proposed that what is more crucial is whether the recipient and theme arguments are pronouns. In this problem, we test these ideas by building simple logistic regression models of the dative alternation to look at the predictive effects of the length and pronominality of the recipient and theme arguments. We will use the `pandas` and `statmodels` Python packages for this.

In general, in studying the factors influencing speaker preference in the dative alternation, we will be interested in estimating the following probabilistic model:

$$P(\text{Construction} = \text{Double Object}|\text{Subject}, \text{Verb}, \text{Recipient}, \text{Theme})$$

where any of a number of features of the subject, verb, recipient, and theme might influence the speaker or writer's choice of linguistic construction. For purposes of this problem, however, we will dramatically simplify. First, we will ignore the subject and verb together, simplifying our problem to:[1]

$$P(\text{Construction} = \text{Double Object}|\text{Recipient}, \text{Theme})$$

Also, we'll only use a few features of the recipient and theme in our predictive model. Recall that logistic regression is characterized by the following equations:

$$\eta = \sum_i \beta_i X_i \qquad \text{(linear predictor)}$$

$$P(\text{success}) = \frac{e^\eta}{1+e^\eta} \qquad \text{(logistic transform of linear predictor)}$$

For the dative alternation, the two possible outcomes are the double object construction (**DO**) and prepositional dative construction (**PD**). We arbitrarily choose DO as the outcome corresponding to"success".

Unlike the case of binomial ordering choice, there is a systematic difference between the possible outcomes that holds across all instances of the dative alternation: it is always the same two constructions that are being chosen between. To capture the possibility of an overall preference for one construction or the other, we add what is called an "intercept" or "bias" term to the equation determining the linear predictor. This is often expressed in the statistics literature (assuming M predictors):

$$\eta = \alpha + \sum_{i=1}^{M} \beta_i X_i \qquad \text{(linear predictor),}$$

but it can equivalently be expressed by defining a "dummy" predictor X_0 whose value is always 1, and writing the linear predictor as:

$$\eta = \sum_{i=0}^{M} \beta_i X_i \qquad \text{(linear predictor),}$$

[1] This is an oversimplification, actually: in particular, the identity of the verb has a *lot* of predictive information. This information is most effectively brought into our model by introducing a hierarchical component like the one we studied in the last part of our in-class coverage of binoimals, endowing verbs with idiosyncratic preferences for which construction they prefer and by how much. This problem doesn't cover these models, but Bresnan et al. (2007) and Morgan and Levy (2015) are good references for seeing how to include this hierarchical (sometimes called "mixed-effects") component for the dative alternation and for binomials, respectively.

a formulation more common in the machine learning literature. (Note that it would be inappropriate to include an intercept in the model for binomial ordering preferences, because there is no intrinsic difference between "success" and "failure" outcomes that is consistently defined across different specific cases of binomial ordering choice.)

We can evaluate the quality of a logistic regression model in a couple of ways. One is predicting the CLASS of the outcome: here, DO or PD? We say that a logistic regression model predicts "success" for a datum if it assigns $P(\text{success}) > 0.5$ for that datum, otherwise "failure". A second is the LOG-LIKELIHOOD of the dataset—the summed log-probabilities of all the observations in the dataset under the fitted model.

Tasks:

1. Define and implement an 80/20/20 train/validation/test random split of the `dative` dataset. (Depending on the version of this problem you receive, you may not use the validation set at all. A validation set is used when it is needed separately from the training set to fully determine the training procedure, e.g., for choosing model hyperparameters or for early stopping to avoid overfitting when training a multi-layer neural network model. For logistic regression fitted via maximum likelihood as we are doing here, it is not necessary.)

2. Fit a logistic regression model to the training set that uses *only* recipient pronominality and an intercept term. What is its classification accuracy on the held-out test dataset? How about its log-likelihood?

3. Add theme pronominality as a predictor to the model and see whether that improves the model's predictive power as assessed by held-out classification accuracy and log-likelihood.

4. Determine whether additionally adding theme and recipient length (in number of words) to the model further improves fit. Try both raw length or log-transformed length. Which gives better performance?

5. Look at the β coefficients of a fitted model with all four predictors and interpret them theoretically (don't worry about interpreting the intercept α, whose value will depend on the numeric coding scheme used for the predictors). Are there any general linguistic principles manifested in the values of all four predictors? Do you see any ways to simplify the model (reduce the number of predictor weights that have to be learned) based on general linguistic principles, without sacrificing much predictive accuracy? This may involve creating a new set of predictors that are a function of the four predictors you've been working with up until now.

2 Logistic Regression versus Single Hidden Layer Neural Networks

For this problem, we will compare the performance of logistic regression versus a single hidden layer neural network in two examples.

The first is the logistic regression problem we looked at earlier. The Colab notebook above contains a pytorch implementation of a single hidden-layer neural network with the same five predictors—a dummy "intercept" input, plus pronominality and length of the recipient and theme—as used in the logistic regression problem. Read through the code and run it. Does the neural network perform any better than the logistic regression model in successfully predicting the dative alternation in the test set? Try a few different values for the hidden layer size and learning rate (the `lr` keyword in `optim.Adam()`) to see if the result is sensitive to that.

The second is example is binary *sentiment classification* on a toy dataset. The task is to classify each sentence as a positive or negative statement based on the words it contains. Here is the training set:

```
It is a good movie. Positive
That movie is good. Positive
This movie is not bad. Positive
It is a movie that is not bad. Positive
This movie is bad. Negative
This is not a good movie. Negative
It is a bad movie. Negative
That movie is not good. Negative
```

Here is the test set:

```
That movie is bad. Negative
That is a movie that is not bad. Positive
That is a good movie. Positive
It is a bad movie. Negative
```

And here is a validation set:

```
That is a movie that is not bad. Positive
That movie is bad. Negative
That is a bad movie. Negative
It is a good movie. Positive
```

Your job: First, featurize each example into a vector of `0/1` "indicator" features, one for each word relevant to the task, where a value of `1` indicates that the word is present in the sentence and a value of `0` indicates that it is not. For simplicity, delete punctuation, convert all words to lower case, and ignore the words *this*, *that*, *it*, *is*, and *a*, so that you are left with the relevant words *movie*, *good*, *bad*, and *not*. So, if your vector has indicator features of those words in that order, the vector for the first example in the training set would be `[1,0,1,0]`.

Then, train a logistic regression model on the training set using this featurization. How well does the resulting trained model work on the test set?

Finally, train a single hidden-layer neural network model for binary classification on the training set, using the validation set to find the best-fitting set of weights. How well does the resulting trained model work on the test set? Is it any better than the logistic regression model? Once again, if necessary play around a bit with the hidden layer size and learning rate to be sure.

How does the relative performance of the neural network vs logistic regression look on the dative-alternation problem versus the sentiment-classification problem? What broader points does it illustrate? Provide your interpretation.

3 Testing the syntactic capabilities of GPT-2

For this problem, you can use code from this Colab notebook, from our in-class exercise on November 8, 2023. https://colab.research.google.com/drive/1OQ0UgQVCMK9kwRweGFcmMj8JpW7hQB_7?usp=sharing

Consider the following sentence:

(3) John picked up his aunt and his uncle picked up grandma.

1. When one first reads this sentence, the second instance of *picked up* is a bit surprising—you may have noticed this when you read it. This has to do with a local syntactic ambiguity earlier in the sentence. Explain (informally is fine) this local syntactic ambiguity and how you might model this surprise effect using a probabilistic generative grammar and incremental parsing.

2. How could the context preceding the second instance of *picked up* be modified to reduce this surprise effect, while preserving the syntactic relations among the words? **Want a challenge?** Think of more than one type of modification that will have this effect.

3. Use the code at the above Colab notebook to test whether the modification you came up with actually reduces the surprisal at *picked up* relative to the original sentence (3) above. Write several versions of the sentence and test each of them, to confirm that the surprisal reduction effect generalizes beyond this specific sentence. Does what you find support the conclusion that GPT-2 makes the syntactic generalizations relevant to this example?

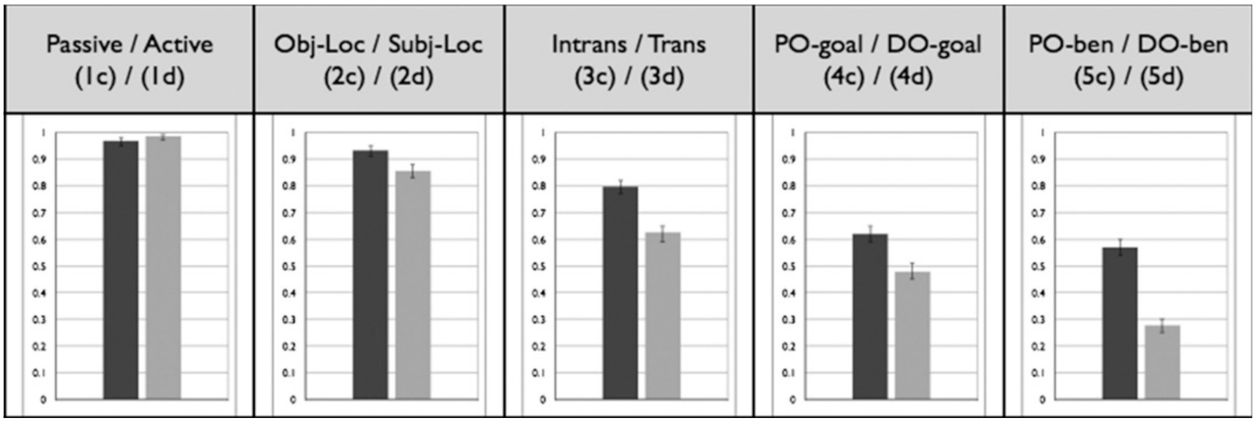

Figure 1: Rates of literal responses to questions in Gibson et al. (2013)

Noisy channel insertions and deletions

Gibson et al. (2013) presented participants with implausible sentences in five pairs of constructions, with one pair exemplified in (3) below:

(3) Transitive/intransitive alternation
a. The tax law benefited from the businessman. (Intransitive)
b. The businessman benefited the tax law. (Transitive)

For each sentence, the researchers asked participants a question that they were supposed to answer on the basis of the presented sentence, for example:

Did the tax law benefit from anything?

If the participant gave an answer consistent with the literal syntax of the sentence (a **yes** answer for this example question), that was considered a "literal" answer; if instead the participant answered in a way that would be consistent with a reassignment of the noun phrase roles to what would be more plausible given commonsense knowledge (for this example, a **no** answer, consistent with the daughter being the recipient and the candle the transferred entity of the giving event), that was considered a "non-literal" answer. Figure 1 shows the **literal-answer response rates** for Example (3) and for the additional construction pairs given below (except for (1), for which there was a null result in the a/b comparison):

(1) Passive/active constructions
a. The girl was kicked by the ball. (Passive)
b. The ball kicked the girl. (Active)

(2) Uninverted/inverted locative constructions
a. The table jumped onto a cat. (Uninverted locative; called "object-locative" in the paper)

b. Onto the table jumped a cat. (Inverted locative; called “subject-locative” in the paper)

(4) Double object/Prepositional phrase object goal constructions

a. The mother gave the daughter to the candle. (PO-goal)
b. The mother gave the candle the daughter. (DO-goal)

(5) Double object/Prepositional phrase object benefactive constructions

a. The cook baked Lucy for a cake. (PO-benefactive)
b. The cook baked a cake Lucy. (DO-benefactive)

You will notice that the “b” version of each pair has a lower literal interpretation rate than the “a” version of the pair (setting aside the Passive/Active contrast, where the literal interpretation rates are nearly 100%). **Question:** within the noisy-channel theory of language comprehension, what is the explanation given for this pattern by Gibson et al.? (**Hint:** think about this in terms of prior and likelihood terms in Bayesian inference.)

References

Baayen, R. H. (2007). The `LanguageR` r package [Available at https://cran.r-project.org/package=languageR

Bresnan, J., Cueni, A., Nikitina, T., & Baayen, H. (2007). Predicting the dative alternation. In G. Boume, I. Kraemer, & J. Zwarts (Eds.), *Cognitive foundations of interpretation* (pp. 69–95). Amsterdam: Royal Netherlands Academy of Science.

Gibson, E., Bergen, L., & Piantadosi, S. T. (2013). Rational integration of noisy evidence and prior semantic expectations in sentence interpretation. *Proceedings of the National Academy of Sciences*, *110*(20), 8051–8056.

Morgan, E., & Levy, R. (2015). Modeling idiosyncratic preferences: How generative knowledge and expression frequency jointly determine language structure, In *Proceedings of the 37th annual meeting of the Cognitive Science Society.*