LING82100: homework 2

(Due 3/6)

1 Reporting a binomial test

As is well known, Mainstream English has two competing forms of the dative.

(1) a. He gave a donation to the museum.
b. He gave the museum a donation.

The former (1a) construction is sometimes called the *prepositional dative* construction, the latter (1b) construction the *double object*. Let us assume, as a null hypothesis, that the two constructions are equiprobable. Bresnan et al. (2007) find, in a large corpus of American English spontaneous phone conversations, 501 prepositional datives and 1,859 double objects. Use the binomial test to test the above null hypothesis.

What to turn in

Provide a short (one or two sentence) "report" of the test. This should include:

- A description of the experiment that allows the reader to infer the values of x and n,
- the test statistic (also called p),
- its 95% confidence intervals, and
- whether the test was significant at $\alpha = .05$.

Also provide a list of the R expression(s) you used to run the test.

Hints

- Ask yourself: what is x, n, and p?
- Consult the help page for `binom.test`, and the notes from last week.

Stretch goal

Using the `dbinom` and `hist` commands, build an attractive histogram of the relative frequencies for the above test. Then add a red vertical line indicating the value of the test statistic. Provide a list of the R expressions you used, and the graph itself. You may want to search online for information about how to save R graphics as PNG image files.

2 McNemar's test

McNemar's test is a type of sign test—itself a special case of the binomial test—commonly used to compare systems in speech & language technology research (e.g., Dror et al. 2018). In McNemar's test, we compute two statistics, x_1 and x_2. one for each of the two systems we are comparing. The first, x_1, is simply the number of "wins" for the first system; that is, it is the number of times the first system makes the correct prediction *and* the second system makes an incorrect prediction. The second, x_2 is the number of "wins" for the second system, and is similarly defined. (We completely ignore cases where both systems make the correct prediction, and where both systems make an incorrect prediction.) we then perform a binomial test with $x = \min(x_1, x_2)$ and $n = x_1 + x_2$, and $p = .5$. The null hypothesis is that the two systems have the same number of "wins", and the alternative hypothesis that one system has more "wins" than the other.

Gorman and Bedrick (2019) use this test and data from the Penn Treebank (PTB) to compare the accuracies of various part-of-speech taggers. The TSV file `PTB.tsv`[1] has some results from Gorman and Bedrick. Load the TSV file into R and compute:

- The number of "wins" for the Stanford tagger (column `Stanford.tag`) over the NLP4J tagger (column `NLP4J.tag`).
- The number of "wins" of the NLP4J tagger over the Stanford tagger.
- The McNemar test results; is one tagger significantly better than the other at $\alpha = .05$? If so, which one?

What to turn in

Provide a list of the R commands you used, and the answers you obtained.

Hints

- You may have to tell the `read.table` function that there is a header row, and not to treat '`#`' as a comment character; consult the help page for more information.
- The column labeled `gold.tag` contains the gold (i.e., true) data.
- To compute the "wins" for the two systems, you may want to use the `!`, `==` and `&` operators and the `sum` function. The following example may help. Given a data frame `d`, the expression:

  ```
  > TnT.correct <- d$gold.tag == d$TnT.tag
  ```

  computes a boolean vector (`TnT.correct`) where the value is `TRUE` if and only if the TnT tagger predicted the correct tag. Let us imagine that we have also computed `Collins.correct`, here the correct/incorrect boolean vector for the Collins tagger. Then, the expression:

[1] `http://wellformedness.com/courses/LING82100/Data/PTB.tsv`

```
> x1 <- sum(TnT.correct & !Collins.correct)
```

computes the number of “wins” wins for TnT over Collins and assigns it to x1. If you get stuck, print the vectors out to confirm they are what you think they are.

References

Bresnan, Joan, Anna Cueni, Tatiana Nikitina, and R. Harald Baayen. 2007. Predicting the dative alternation. In *Cognitive Foundations of Interpretation*, 69–94. Amsterdam: KNAW.

Dror, Rotem, Gil Baumer, Segev Shlomov, and Roi Reichart. 2018. The hitchhiker’s guide to testing statistical significance in natural language processing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1383–1392. Melbourne: Association for Computational Linguistics.

Gorman, Kyle, and Steven Bedrick. 2019. We need to talk about standard splits. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2786–2791. Florence: Association for Computational Linguistics.