Juliana McCausland
Ling 473
Project 5

**Results**:

```
1     The scope of the discipline of statistics broadened in the early 19th century to include the
collection and analysis of data in general.
tgl   -84.69652345966465
pol   -86.60758590143108
spa   -86.65472097295297
dut   -66.15529491498913
nob   -87.72935459124987
dan   -86.34014822169614
fin   -90.37280726773606
fra   -90.37280726773606
eng   -9.145181479292324
ita   -76.44247884651136
swh   -87.32165474528868
swe   -82.7335878022329
deu   -79.81330004409632
gla   -85.67231123106775
por   -87.13200849661874
result eng
```

**Description**:

I may be the only person who really struggled with this project compared to the prior projects. I had to constantly reframe my math approach because of constant errors despite having written it out in a way that seemed intuitive. Overall the program seems to be okay, but it is not 100% accurate.

The program opens the language model files using the latin_1 encoding, extracts the language names, words, and the word counts. The words are added to a nested dictionary in which each word contains the counts corresponding to each of the 15 languages. The test file is then opened, cleaned using the translate() tool to strip punctuation, and then the probability function is called. I used add-1 smoothing (or some version of that, which was mentioned in the Jurafsky book). The probabilities are calculated log base 10(counts for each word given each language/total counts for that word). If a word was not in the dictionary, it got the same probability as a “singleton” (as mentioned in spec).

Based on the arguments in run.sh/run-extra.sh, the program determines if it is using the extra credit data or not. If not, results are printed using the appropriate format. For extra credit, there is a threshold of 15 for deciding if a language is unknown or not. I used 15 based on the inaccuracies I saw in the normal output. If the difference is greater than 15, results are printed using the appropriate format.

A character error was fixed (with Sheng’s help in canvas) by adding the line: PYTHONIOENCODING=utf8 to my run files.
I think my biggest obstacle was my ignorance of programming and python. I constantly found myself debugging python-specific things and trying to figure out how to do basic math in python (I’m not sure why I struggled with this, but mental exhaustion may have contributed a tiny bit). I plan on spending my free time fixing up this code and getting more comfortable with these concepts after the course is finished!