Caitlyn Jones

Prof. Yangyang Wang

MATH 40A: Intro to Applied Math

2 October 2024

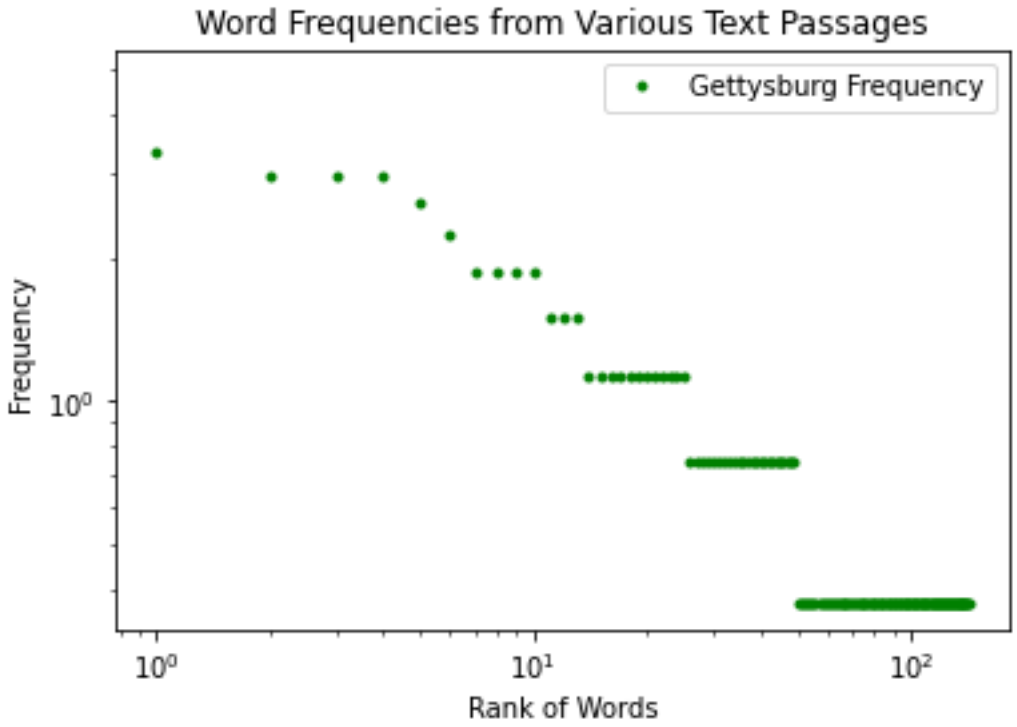
Lab 2

1. To plot the results of the frequency distribution of the Gettysburg address, I first called the function `wordcount(filename)` on the text file gettysburg.txt.

```
results0, freq0 = wordcount("gettysburg.txt")
```

Then, I plotted the frequency results on a graph with labels, axes, and a legend.

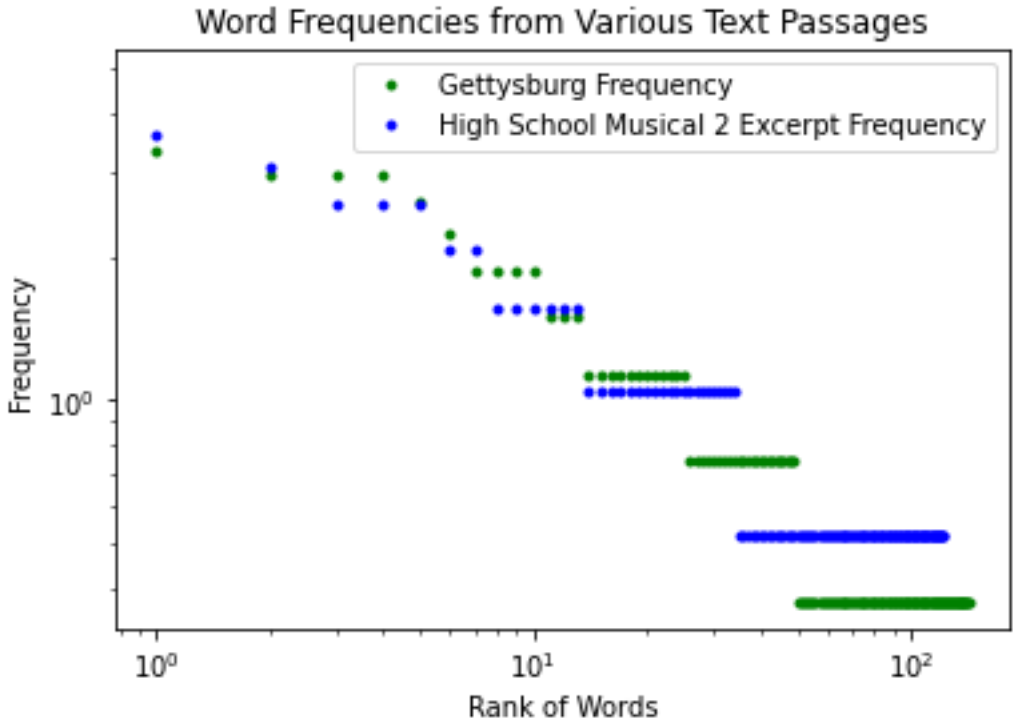
```
p0 = plt.loglog(freq0, 'g.', label = "Gettysburg Frequency")
plt.title("Word Frequencies from Various Text Passages")
plt.xlabel("Rank of Words")
plt.ylabel("Frequency")
plt.legend()
```



2. For my random text file, I chose an excerpt from my favorite movie *High School Musical 2*. I followed the same steps as before to compute frequency and then graphed it on the same graph as before.

```
results1 , freq1 = wordcount("hsm2.txt")
```

```
p1 = plt.loglog(freq1, "b.", label = "High School Musical 2 Excerpt Frequency")
plt.title("Word Frequencies from Various Text Passages")
plt.xlabel("Rank of Words")
plt.ylabel("Frequency")
plt.legend()
```



The top words in both passages are similar, as they are all articles or very common words in the English language. The differences arise when the words become more specific to the passage.

3. My data does not agree with Zipf's law, as the most frequent words appear roughly the same amount of time, give or take a few instances.

4. To compute the entropy based on the equation $H = -\sum_{i=1}^{M} f_i log_2(f_i)$, I created a function `entropy(frequency)` that takes in the frequency from the output of `wordcount(filename)` and first assigns that value to f. Then, I made a variable `coeff` that computes the coefficient in the summation formula. Finally, I created a variable `summation` that will hold the computed sum of the equation.

```
def entropy(frequency):
    f = frequency
    coeff = f * np.log2(f)
    summation = -np.sum(coeff)
    return summation
```

5. To measure the entropy of each text, I called the function entropy(frequency) on freq0 and freq1, which were the results from calling wordcount(filename) on both texts.

```
print("Entropy of Gettysburg Address: ", entropy(freq0))
print("Entropy of High School Musical 2 Excerpt: ", entropy(freq1))
```

```
Entropy of Gettysburg Address:  0.08158556511866713
Entropy of High School Musical 2 Excerpt:  -0.010201385190593775
```

The Gettysburg Address has a higher information content because it has a higher entropy.