Advanced Topics in Computer Science Activity – Zipf's Law

BACKGROUND

Zipf's Law is an interesting phenomenon related to the relative frequency of words in a language. Begin with the most popular in any language (*the* in English, for example). Zipf's Law predicts that the *second* most popular word will be one-*half* $\left(\frac{1}{2}\right)$ as common as the first word. The *third* most popular word will be one-*third* $\left(\frac{1}{3}\right)$ as common as the first word. The fourth most popular word will be one-fourth as common as the first word, and so on.

You can learn more about Zipf's Law by reading Wikipedia and watching YouTube (see URLs in the *References* section below).

OBJECTIVE

In this assignment you'll be writing a program to count the words in a large body of text and determine whether or not the words in that text closely follow Zipf's Law.[1]

DELIVERABLES

zipf_analysis.zip

This zipped directory will contain `zipf_analysis.py` and the text file that you used to create your word frequency counts.

PROCEDURE

Write a program `zipf_analysis.py` that:

1. imports a text file containing a large number of English words.

2. uses a dictionary to perform a word-frequency count: identify how many times the word *the* appears, how many time the word *and* appears, etc., for every word in the text file.

3. includes a Docstring comment at the beginning of the source file, with a one-to-two paragraph description of:
   a. what your program does
   b. its results (word frequency)
   c. what Zipf's Law is
   d. whether you think your results tend to *support* or *refute* Zipf's Law, and why

ASSIGNMENT NOTES

1. You'll need to obtain a text file containing a representative sample of a large number of words,

[1] Thanks to Joe and Marilyn Zeronian for inspiration in the development of this activity.

probably in the English language. Considering downloading a UTF-8 text file from Project Gutenberg (URL in *References* below) or using a cleaned up file of words from the instructor.

2. Read the lines/words from that file and get them into a Python `list` structure.

3. Set up an empty Python `dict` structure that you can use to store the words and their frequencies as you count them.

4. Use one of these code snippets to count the words in your list into the dictionary structure:
   for word in words:

   ```
   word_counts = {}
   if word in word_counts.keys():
       word_count[word] += 1
   else:
       word_counts[word] = 1
   ```

   ... or ...

   ```
   word_counts = {}
   for word in words:
       word_counts[word] = word_counts.get(word, 0) + 1
   ```

5. Based on your analysis, print out at least the ten most popular words and their frequency. A Python dictionary is convenient for counting words, but the collection of key-pair values doesn't allow for ordering, so you'll want to convert your unordered dictionary to a list that can be sorted.

   ```
   import operator
   .
   .
   .
   sortedwords = sorted(word_counts.items(),
   key=operator.itemgetter(1))
   ```

6. Your Docstring comment at the beginning of the program will include an extensive, 1-2 paragraph explanation of your program, your results, what Zipf's Law is, and how your programs results reflect on Zipf's Law.

QUESTIONS FOR YOU TO CONSIDER (NOT HAND IN)

1. Zipf's Law is not a "scientific law" but an "empirical statistical law." What is the difference between the two types of laws?

2. Zipf's Law has been proposed to apply to other domains/contexts beyond linguistics. What are some of these other domains? Does it make sense to you that Zipf's Law might apply in these other contexts? Is there quantitative evidence that Zipf's Law *does* apply in these cases?

3. Zipf's Law relationships are often displayed using a *log-log* graph. What is a log-log graph, and why is it appropriate for examining Zipf's Law?

REFERENCES

Project Gutenberg (http://www.gutenberg.org) - source of text files

Zipf's Law (Wikipedia, https://en.wikipedia.org/wiki/Zipf%27s_law)

The Zipf Mystery (YouTube, https://www.youtube.com/watch?v=fCn8zs912OE)

T*he U.S. Megalopolis Isn't as Politically Powerful as You Think* (blogpost, http://www.realclearpolitics.com/articles/2017/02/06/the_us_megalopolis_isnt_as_politically_powerful_as_you_think_132990.html)