

assignment1

September 15, 2019

1 Assignment 1

John Flanigan

September 16, 2019

```
[1]: from nltk.book import *
```

```
*** Introductory Examples for the NLTK Book ***
Loading text1, ..., text9 and sent1, ..., sent9
Type the name of the text or sentence to view it.
Type: 'texts()' or 'sents()' to list the materials.
text1: Moby Dick by Herman Melville 1851
text2: Sense and Sensibility by Jane Austen 1811
text3: The Book of Genesis
text4: Inaugural Address Corpus
text5: Chat Corpus
text6: Monty Python and the Holy Grail
text7: Wall Street Journal
text8: Personals Corpus
text9: The Man Who Was Thursday by G . K . Chesterton 1908
```

1.1 Exercise 1.5

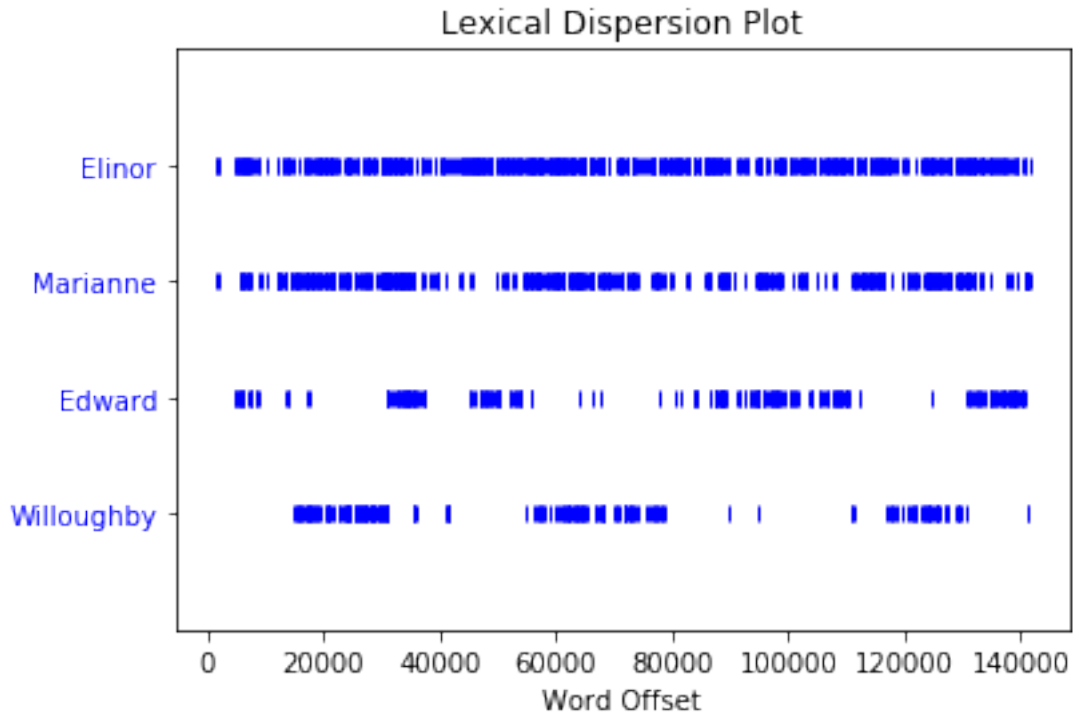
Compare the lexical diversity scores for humor and romance fiction in 1.1. Which genre is more lexically diverse?

Humor had a lexical diversity of 0.231 and romance fiction had a lexical diversity of 0.121 meaning humor is more lexically diverse.

1.2 Exercise 1.6

Produce a dispersion plot of the four main protagonists in *Sense and Sensibility*: Elinor, Marianne, Edward, and Willoughby. What can you observe about the different roles played by the males and females in this novel? Can you identify the couples?

```
[3]: text2.dispersion_plot(["Elinor", "Marianne", "Edward", "Willoughby"])
```



Based on the lexical dispersion plot above, Elinor and Marianne appear more frequently and consistently throughout the novel. The two male characters, Edward and Willoughby, appear sporadically. If we assume that the couples are composed of a male character and a female character, we can identify one couple to be Marianne and Willoughby because Marianne appears more frequently whenever Willoughby is appearing. This leaves Elinor and Edward as the other couple.

1.3 Exercise 1.27

Define a function called `vocab_size(text)` that has a single parameter for the text, and which returns the vocabulary size of the text.

```
[4]: def vocab_size(text):
      return len(set(text))

[5]: # Example execution of vocab_size
      print('Vocabulary size of text 1:', vocab_size(text1))
      print('Vocabulary size of text 3:', vocab_size(text3))
```

```
Vocabulary size of text 1: 19317
Vocabulary size of text 3: 2789
```

1.4 Exercise 1.28

Define a function `percent(word, text)` that calculates how often a given word occurs in a text, and expresses the result as a percentage.

```
[6]: def percent(word, text):  
      count = text.count(word)  
      total = len(text)  
      return 100 * count / total  
  
[7]: # Test percent by comparing characters in Sense and Sensibility  
      print('Percent Marianne:', percent('Marianne', text2))  
      print('Percent Willoughby:', percent('Willoughby', text2))
```

Percent Marianne: 0.39978527434028366
Percent Willoughby: 0.15186189749675086