Frequency Analysis

- The use of frequency analysis as a tool for cryptanalysis has its origins in ancient Arab civilizations, namely the work of the Arabic scientist Al-Kindi.

- Al-Kindi authored works on a number of important mathematical subjects, including arithmetic, geometry, the Indian numbers, the harmony of numbers, lines and multiplication with numbers, relative quantities, measuring proportion and time, and numerical procedures and cancellation.

- He also wrote four volumes, *On the Use of the Indian Numerals* (Ketab fi Isti'mal al-'Adad al-Hindi) which contributed greatly to diffusion of the Indian system of numeration in the Middle-East and the West. In geometry, among other works, he wrote on the theory of parallels. Also related to geometry were two works on optics.

- Al-Kindi is credited with developing a method whereby variations in the frequency of the occurrence of letters could be analyzed and exploited to break ciphers (i.e. cryptanalysis by frequency analysis).

- His book on this topic is *Istikhraj al-Kotob Al-Mu'amah* الـمعماة الـ كـ تب ا سـ تخراج فـ ي ر سالـة (Literally: *On Extracting Obscured Correspondence*, more contemporary: *On Decrypting Encrypted Correspondence*).

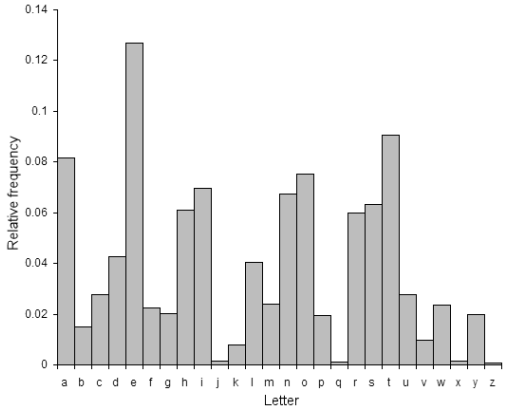
- The flowing is the first page of Al-Kindi's manuscript "On Deciphering Cryptographic Messages", containing the oldest known description of cryptanalysis by frequency analysis:

- The following are translated excerpts from his book entitled "*A Manuscript on Deciphering Cryptographic Messages*"s:

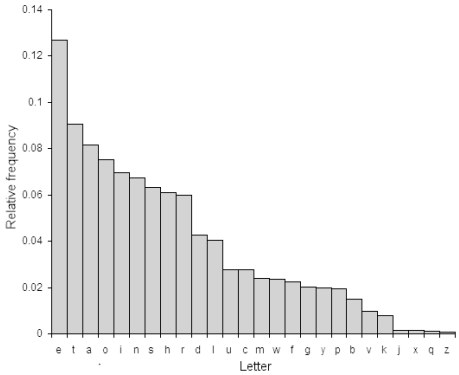
"One way to solve an encrypted message, if we know its language, is to find a different plaintext of the same language long enough to fill one sheet or so, and then we count the occurrences of each letter. We call the most frequently occurring letter the 'first', the next most occurring letter the 'second', the following most occurring the 'third', and so on, until we account for all the different letters in the plaintext sample".

"Then we look at the cipher text we want to solve and we also classify its symbols. We find the most occurring symbol and change it to the form of the 'first' letter of the plaintext sample, the next most common symbol is changed to the form of the 'second' letter, and so on, until we account for all symbols of the cryptogram we want to solve"

- We use exactly the same process in modern times when conducting frequency analysis as part of cryptanalysis. Consider the following graphs presented earlier, which illustrate the relative distributions of the 26 letters in an average English text:



  - The following chart shows the letter frequencies ordered from most frequent to least frequent:

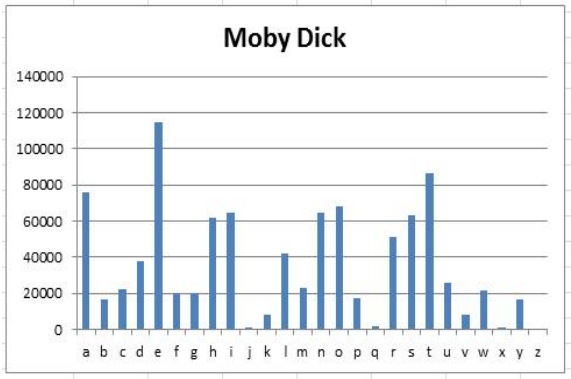


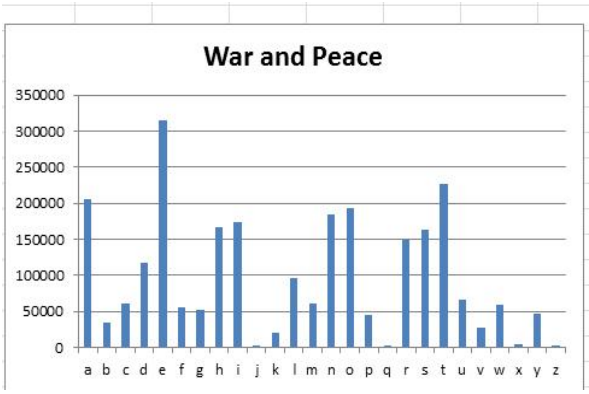
  - As we can see, the letter 'e' is the most common, and appears almost 13% of the time, whereas 'z' appears far less than 1 percent of time.

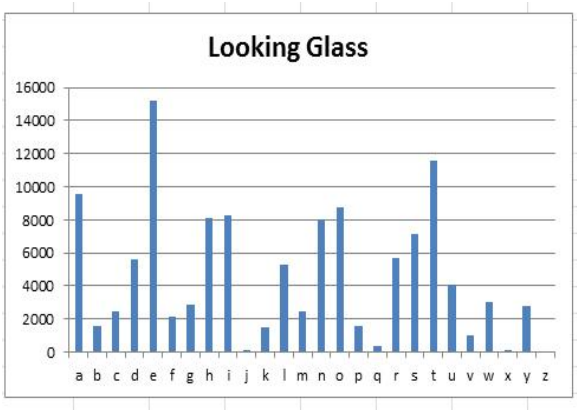
- The sequence "**ETAOIN**" consisting of the six most frequent letters is a very useful one in the cryptanalysis of text-based cipher such as substitution and transposition ciphers.

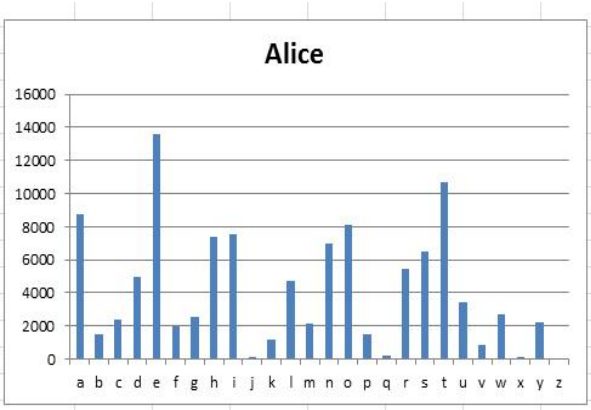
- We can in fact generate these distributions by taking and English text or work of literature to verify that these distributions are very similar in the English language.

- Consider the following graphs that illustrate the English letter distributions from four works of literature:









- Notice the very strong statistical correlations in these distributions. The point is that frequency analysis (for specific languages) is a very powerful tool when breaking substitution and transposition ciphers.

- Messages encrypted with such ciphers contain all the original letters of the original English plaintext, except in a different order. However, the **frequency** of each letter in the ciphertext remains the same: E, T, and A should occur much more often than Q and Z.

- We can also apply this technique to the Vigenère cipher, it essentially uses multiple Caesar cipher alphabets (and a keyword) to encrypt a plaintext block.

- Thus we can use frequency analysis to break each subkey one at a time based on the letter frequency of the attempted decryptions.

- Instead of performing English word detection on the ciphertext, we analyze the letter frequency of each subkey's decrypted text.

- By matching the letter frequency of regular English, we could attempt try several different algorithms. For example, we can order the letters from most to least frequent and calculate a **metric** (**frequency match**) or score for this ordering of frequencies.

- To calculate the frequency match score for a string, the score starts at 0 and each time one of the letters **E, T, A, O, I, N** appears among the six most frequent letters of the string, we add a point to the score.

- And each time one of the letters **V, K, J, X, Q, Z** appears among the six least frequent letters of the string, we add a point to the score.

- The score for a string will be an integer from 0 (meaning the letter frequency of the string is completely unlike regular English's letter frequency) to 12 (meaning it is identical to regular English's letter frequency).

- Breaking the Vigenère cipher is much more complicated than breaking substitution or transposition ciphers. The first step is to decrypt the letters for the first letter of the keyword with each of the 26 possible letters and find out which decrypted ciphertext produces a letter frequency that matches English most closely, which is a good indication that the correct letter has been locate.

- This process is repeated for every letter in the keyword. Note part of the complexity is also determining the length of keyword, which will be discussed later.

- Assuming a six-letter keyword, we need only perform 26 decryptions for each subkey individually, and the machine computer only has to perform (26)(6) or 156 decryptions as opposed to 11,881,376 decryptions.

- A frequency analysis module for this purpose will require the following minimum functionality:

    - A function will take a string parameter and return a dictionary that has the count of how often each letter appears in the string.
    - A function will take a string parameter and return a string of the 26 letters ordered from those that appear most frequently to least frequently in the string parameter.
    - This function will take a string parameter and return an integer from 0 to 12 of the string's letter frequency match score.

- You have already designed an application for the functionality described above. This will be used later to break the Vigenère cipher.