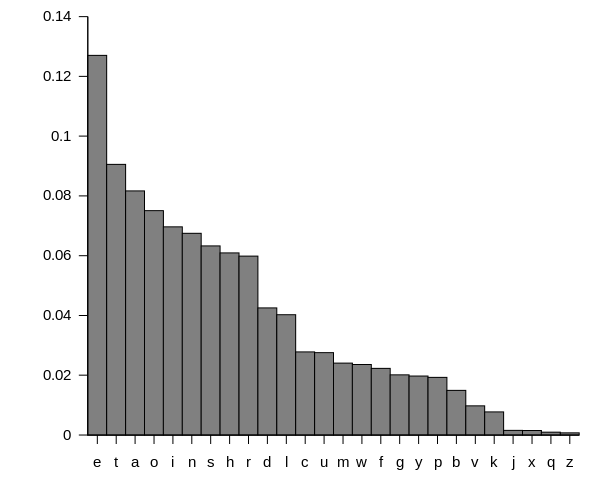
**CSC 3621 Coursework 1**

**Exercise 1 Report**

The goal of this exercise was to use a given text file to decrypt an encoded passage of text using a frequency cypher. The program I wrote reads in a text file, then counts the occurrences of each letter and orders them by frequency. Reading in the input file produced the following results(left)



When compared to a graph of commonly known letter frequencies(right) (source: <https://commons.wikimedia.org/w/index.php?curid=9971079>) the trend is similar, however certain letters are switched around, which may be due to the choice of input text, a Sherlock Holmes story which contains fairly archaic language. Furthermore, the longer the text the more accurate the frequency analysis will be, so if the text file was longer then it may have been closer to the commonly accepted one.

When the cypher text is also read into the frequency analysis, it gives the results displayed below. As can be seen, these results are a lot more inconclusive, considering the small size of the text to decrypt- certain letters (such as g and y) had the same number of occurrences, which meant the substitution would be inaccurate.

The next step is to substitute the most frequently occurring letter in the cypher text with the most frequently occurring letter in the input text, for example e for i. When this is done for all 26 letters, the cipher text reads as follows:

"yheuioedg."

"tren gsm wag entihedg hedg sn we."

"trat io exueddent. i trink, yehrayo, it io adwsot tiwe trat i

yheyahe psh tre nec hsde i rave ts ydag."

re lioayyeahel ints rio belhssw anl hetmhnel in a pec winmteo in

tre urahauteh sp an awiabde anl oiwyde-winlel nsnusnpshwiot

udehfgwan. rio bhsal bdauk rat, rio baffg thsmoeho, rio crite

tie, rio ogwyatretiu owide, anl fenehad dssk sp yeehinf anl

benevsdent umhisoitg cehe omur ao wh. qsrn rahe adsne usmdl rave

ejmaddel. it cao nst wehedg trat rsdweo uranfel rio usotmwe. rio

exyheooisn, rio wanneh, rio vehg osmd oeewel ts vahg citr evehg

pheor yaht trat re aoomwel. tre otafe dsot a pine autsh, even ao

ouienue dsot an aumte heaosneh, cren re beuawe a oyeuiadiot in

uhiwe.

it cao a jmahteh yaot oix cren ce dept bakeh otheet, anl it otidd

cantel ten winmteo ts tre rsmh cren ce psmnl smhoedveo in

oehyentine avenme. it cao adhealg lmok, anl tre dawyo cehe qmot

beinf difrtel ao ce yauel my anl lscn in phsnt sp bhisng dslfe,

caitinf psh tre uswinf sp ito suumyant. tre rsmoe cao qmot omur

ao i ral yiutmhel it phsw orehdsuk rsdweo' omuuinut leouhiytisn,

bmt tre dsuaditg ayyeahel ts be deoo yhivate tran i exyeutel. sn

tre usnthahg, psh a owadd otheet in a jmiet neifrbsmhrssl, it cao

hewahkabdg aniwatel. trehe cao a fhsmy sp orabbidg lheooel wen

owskinf anl damfrinf in a ushneh, a ouioosho-fhinleh citr rio

creed, tcs fmahlowen crs cehe pdihtinf citr a nmhoe-fihd, anl

oevehad cedd-lheooel gsmnf wen crs cehe dsmnfinf my anl lscn citr

uifaho in treih wsmtro.

"gsm oee," hewahkel rsdweo, ao ce yauel ts anl phs in phsnt sp

tre rsmoe, "trio wahhiafe hatreh oiwydipieo watteho. tre

yrstsfhayr beusweo a lsmbde-elfel ceaysn nsc. tre uranueo ahe

trat ore csmdl be ao avehoe ts ito beinf oeen bg wh. fslpheg

nshtsn, ao smh udient io ts ito uswinf ts tre egeo sp rio

yhinueoo. nsc tre jmeotisn io, crehe ahe ce ts pinl tre

yrstsfhayr?"

"crehe, inleel?"

Whilst this is clearly not correctly decrypted, it is in a good enough state to begin manual tuning. A function in my program takes characters from the command line and swaps them, converting between upper and lower case to ensure letters are not swapped transitively and mistakes made that way. The following swaps need to be made:

* D and L
* H and R
* C and W
* S and O
* C and U
* Y and P
* U and M
* G and Y
* G and F
* Q and J

Once these ten swaps have been made, the cypher text is decrypted.

If I were to repeat this exercise, I would omit the need for input text entirely and hard code in the frequency of letters as commonly accepted. Not only does this save on computational power, it requires less manual swaps to be performed (seven compared to ten) allowing a result to be obtained quicker.