Benjamin Steenkamer

CPEG 672-010

Project 1: Blackhat challenge

**1) What is your project?**

- The project is to break 5 historical ciphers. I have 5 Python files that attempt to decode the cipher text. Some of the methods work generically for any cipher text, and some are specilized to work only for the given cipher text.

**2) Where can I run it? (Do I need some files downloaded?)**

- There are 5 Python 3 source code files you can run. I have solved all the ciphers except the Playfair cipher. If you run python3 <file>.py, it should print out the decoded text. For playfair.py, it will instead launch a shotgun hill climb analysis and attempt to find a potential key. For the ciphers that I broke, you can uncomment the some of the lines to shows the cracking process at work instead of just showing the final result.

The folder called encrypted contains the cipher text from the emails. The output folder is where hill2x2.py will output its results, instead of printing them to the terminal.

**3) Who did you work with? If part of a group, how would you describe your role?**

- This was a single person project. I did not work with anyone.

**4) What grade would you give yourself (out of 100)?**

**5) How many hours did you work on this?**

**6) List your "a-ha" moments. (When something useful clicked for you.)**

**7) What parts of your project do you wish to improve?**

**8) What is your overall impression of the experience? Which skills do you want to keep in your future?**

**Notes I took during the decoding process:**

**1.txt**:

Cipher used: Vigenere

key length = 16

Key used decode text: [1, 4, 20, 22, 20, 11, 7, 14, 5, 6, 24, 14, 13, 11, 8, 7]

Decoded text: itisintheuncompromisingnesswithwhichdogmaisheldandnotinthedogmaorwantofdogmathatthedangerliesthepowerofdoinganythingwithquicknessisalwaysprizedmuchbythepossessorandoftenwithoutanyattentiontotheimperfectionoftheperformancethelifeofeverymanisadiaryinwhichhemeanstowriteonestoryandwritesanotherandhishumblesthouriswhenhecomparesthevolumeasitiswithwhathevowedtomakeithonestpeopledonthidetheirdeedsitisanoldmaximofminethatwhenyouhaveexcludedtheimpossiblewhateverremainshoweverimprobablemustbethetruthknavesdogrowgreatbybeinggreatmensapesignoranceistheparentoffearcourageisfireandbullyingissmoken

**2.txt**:

Cipher used: Playfair since it has no j's and and the observed vs english squared sums are not equal

Observed Frequency Sums Squared = 0.051173924229035474

English Frequency Sums Squared = 0.0439782820612771

Has no j's

After running the shotgun hill climb function for several hours the best board found was:

dtkue

snpfc

wgzhr

laimo

ybqxv

It decodes the cipher text into text with a squared English frequency sum error of: 0.0027312374492883887

I would need to run this script for a very long time for it to find a key close enough to the actual key. I could then reverse engineer the key by looking for partially completed words in the decoded text and make changes to the key so that the words would be completed.

**3.txt**:

Cipher used: Column transposition cipher because the observed vs. English frequencies are the same and close to 0.065

Observed Frequency Sums Squared = 0.06444444444444444

English Frequency Sums Squared = 0.0643784115158248

Has no j's

Key size = 8, 9, or 10 columns; k is a permutation of the 8, 9, or 10 columns

Took 2m56.346s brute force all 3628800 column transpositions using 22 processes

k = 10

Column transposition: (6, 8, 2, 4, 1, 7, 3, 5, 9, 0)

Decoded text: itsnothingreturnedmrschickitsmerelychangeofweatherwemustexpectchangenooneisevertoooldtodoafoolishthingunwelcometruthsarenotpopularyetthefirstbringerofunwelcomenewshathbutalosingofficeandhistonguesoundseverafterasasullenbellremembredtollingadepartingfriendwillnotatinyspeckveryclosetoourvisionblotoutthegloryoftheworldandleaveonlyamarginbywhichweseetheblotmostofthetroubleinlifecomesfrommisunderstandingithinksaidannewouldtheworldeverhavebeenmadeifitsmakerhadbeenafraidofmakingtroublemakinglifemeansmakingtroubletobeconsciousthatyouareignorantisagreatsteptoknowledgemadnessingreatonesmustnotunwatchdgo

**4.txt**:

Cipher used: Substitution cipher, because the observed frequency sums squared is fairly close to actual English.

Observed Frequency Sums Squared = 0.06984356253421704

English Frequency Sums Squared = 0.03652875472329036

OBS = [('d', 0.1296928327645051), ('z', 0.1075085324232082), ('v', 0.08703071672354949), ('c', 0.08532423208191127), ('t', 0.07849829351535836), ('f', 0.06655290102389079), ('a', 0.05802047781569966), ('q', 0.05460750853242321), ('g', 0.04436860068259386), ('y', 0.042662116040955635), ('e', 0.032423208191126277), ('p', 0.032423208191126277), ('j', 0.02901023890784983), ('h', 0.025597269624573378), ('s', 0.023890784982935155), ('n', 0.023890784982935155), ('m', 0.020477815699658702), ('b', 0.01877133105802048), ('l', 0.011945392491467578), ('i', 0.010238907849829351), ('o', 0.010238907849829351), ('r', 0.0034129692832764505), ('u', 0.0017064846416382253), ('x', 0.0017064846416382253), ('w', 0.0), ('k', 0.0)]

ENG = [('e', 0.1288623426065769), ('t', 0.09024664994930598), ('a', 0.08064249900208098), ('o', 0.07378315126621263), ('n', 0.06984975410235668), ('i', 0.06905550211598431), ('s', 0.063817324270356), ('r', 0.06156572691936394), ('h', 0.06098726796371807), ('d', 0.04328667139002636), ('l', 0.04101676132771116), ('u', 0.0278568510204016), ('c', 0.026892340312538593), ('m', 0.02500971934780021), ('f', 0.0244847137116921), ('w', 0.021192261444145363), ('g', 0.019625534749730816), ('y', 0.01806326249861108), ('p', 0.017031440203182008), ('b', 0.015373768624831691), ('v', 0.010257964235274787), ('k', 0.006252182367878119), ('x', 0.0016941732664605912), ('j', 0.0011176940633901926), ('q', 0.0010648594165322703), ('z', 0.0009695838238376564)]

OBS\_bigrams\_freq = [('tv', 0.03076923076923077), ('zy', 0.02905982905982906), ('vg', 0.022222222222222223), ('ca', 0.018803418803418803), ('qv', 0.018803418803418803), ('dt', 0.017094017094017096), ('da', 0.017094017094017096), ('ad', 0.015384615384615385), ('yd', 0.015384615384615385), ('vz', 0.013675213675213675), ('qz', 0.013675213675213675), ('dj', 0.011965811965811967), ('yt', 0.011965811965811967), ('zt', 0.011965811965811967), ('fc', 0.011965811965811967), ('cp', 0.011965811965811967), ('vd', 0.011965811965811967), ('gd', 0.011965811965811967), ('fd', 0.011965811965811967),('fz', 0.011965811965811967)]

ENG\_bigrams\_freq = [('th', 0.03882543), ('he', 0.03681391), ('in', 0.02283899), ('er', 0.02178042), ('an', 0.0214046), ('re', 0.01749394), ('nd', 0.01571977), ('on', 0.01418244), ('en', 0.01383239), ('at', 0.01335523), ('ou', 0.01285484), ('ed', 0.01275779), ('ha', 0.01274742), ('to', 0.01169655), ('or', 0.01151094), ('it', 0.01134891), ('is', 0.01109877), ('hi', 0.01092302), ('es', 0.01092301),('ng', 0.01053385)]

OBS\_trigrams\_freq = [('tvg', 0.015410958904109588), ('zyd', 0.0136986301369863), ('qvn', 0.011986301369863013), ('ztv', 0.008561643835616438), ('eca', 0.008561643835616438), ('dta', 0.008561643835616438), ('vgf', 0.00684931506849315), ('tvs', 0.00684931506849315),('dje', 0.00684931506849315), ('ytv', 0.00684931506849315), ('vmd', 0.00684931506849315), ('zyt', 0.00684931506849315), ('fdj', 0.00684931506849315), ('caz', 0.00684931506849315), ('chd', 0.005136986301369863), ('yqv', 0.005136986301369863), ('cvd', 0.005136986301369863), ('yda', 0.005136986301369863), ('cpv', 0.005136986301369863), ('vgz', 0.005136986301369863)]

ENG\_trigrams\_freq = [('the', 0.0181), ('and', 0.0073), ('ing', 0.0072), ('ent', 0.0042), ('ion', 0.0042), ('for', 0.0034)]

Duplicate letter frequencies: [('f', 4), ('z', 4), ('d', 3), ('b', 2), ('e', 2), ('g', 2), ('j', 1), ('m', 1), ('c', 1), ('v', 0), ('l', 0), ('r', 0), ('a', 0), ('t', 0), ('p', 0), ('s', 0), ('k', 0), ('x', 0), ('y', 0), ('i', 0), ('q', 0), ('h', 0), ('o', 0), ('n', 0), ('w', 0), ('u', 0)]

ss, ee, tt, ff, ll, mm, oo

Cipher text: ['a','b','c','d','e','f','g','h','i','j', k','l','m','n','o','p','q','r','s','t','u','v','w','x','y','z']

Substitutions: ['r','p','o','e','f','s','d','m','b','l','k','w','c','g','v','u','i','j','y','a','z','n','q','x','h','t']

Decoded text: dontmindanythinganyonetellsyouaboutanyoneelsejudgeeveryoneandeverythingforyourselfitissafertoacceptanychancethatoffersitselfandextemporizeaproceduretofititthantogetagoodplanmaturedandwaitforachanceofusingittreacheryandviolencearespearspointedatbothendstheywoundthosewhoresorttothemworsethantheirenemiessomehappytalentandsomefortunateopportunitymayformthetwosidesoftheladderonwhichsomemenmountbuttheroundsofthatladdermustbemadeofstufftostandwearandtearandthereisnosubstituteforthoroughgoingardentandsincereearnestnessselflovemyliegeisnotsovileasinasselfneglectingjustbreathingisntlivingp

Using the frequencies of letters, bigrams, and trigrams, I was able to guess the substitutions for the most common letters. From there, I looked at the output and tried to form words using the know letters and filling in the blanks. Once words started forming it, it got easier to guess the substitutions.

**5.txt**:

Cipher used: Hill 2x2, by process of elimination

Observed Frequency Sums Squared = 0.04494655004859088

English Frequency Sums Squared = 0.03790484195679802

<http://crypto.interactive-maths.com/hill-cipher.html>

I used parallel processes to make the brute force go much faster. I originally used Python threading, but then I remember Python threads don’t run in parallel. I changed the method to processes, it gave me the speedup I wanted.

42m37.884s to brute force all 358800 keys with a single process

2m23.459s to brute force all 358800 keys with a 24 processes

Tested key:

[[15 13]

[ 9 24]]

English Frequency Sums Squared: 0.06564966208695856

Decoded text: fairspeechmayhideafoulheartscatteredwitstakealongtimepickingupwhenyouvelearnedtolaughatthethingsthatshouldbelaughedatandnottolaughatthosethatshouldntyouvegotwisdomandunderstandingtrulytoenjoybodilywarmthsomesmallpartofyoumustbecoldforthereisnoqualityinthisworldthatisnotwhatitismerelybycontrastnothingexistsinitselftobevestedwithenormousauthorityisafinethingbuttohavetheonlookingworldconsenttoitisafinerblunttoolsaresometimesfoundofusewheresharperinstrumentswouldfailinheritedideasareacuriousthingandinterestingtoobserveandexaminehappyaretheythatheartheirdetractionsandcanputthemtomending