FCTUC FACULDADE DE CIÊNCIAS
E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Teoria da Informação

Trabalho Prático nº1

Entropia, Redundância e Informação Mútua

Trabalho realizado por:

- João Luís Ambrósio Limeiro 2017262387
- Etiandro Cirilo André António 2017290285
- João Pedro de Sá Dinis 2017248548

1.

Na primeira pergunta, era pedido para escrevermos uma rotina em Matlab, o "histograma.m", que dada uma fonte, tínhamos que determinar e visualizar o histograma de ocorrência dos seus símbolos.

Como havia 3 tipos de ficheiros possíveis (.bmp, .wav e .txt), separámos pelo tipo de ficheiro.

Para as imagens e música, determinámos o alfabeto de cada fonte e usando a função "histogram()", com os parâmetros, fonte e alfabeto criámos o histograma.

Para o texto, com a fonte determinámos o alfabeto e usando a função "bar()" com a quantidade de vezes que as letras apareciam no ficheiro texto como parâmetro .

2.

Nesta pergunta, era pedido para determinarmos a entropia sabendo a fonte.

Escrevemos uma rotina com o nome "calculaEntropia.m" com a fonte como parâmetro, determinamos o alfabeto da fonte e quantas vezes os símbolos do alfabeto se repetiam.

Com estes dois dados calculamos a probabilidade de cada um e determinamos a entropia com as seguintes fórmulas.

$$H=-\sum_{i=1}^{n} p(a_i) log_2 p(a_i)$$

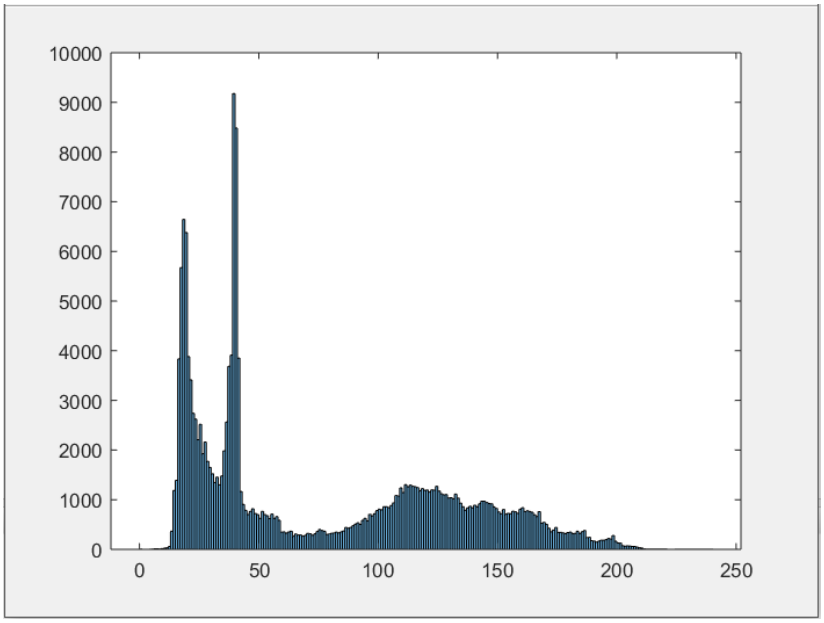
```
-sum(p(p~=0).*log2(p(p~=0)))
```

A primeira é a fórmula da entropia e a segunda é a fórmula da entropia mas adaptada para o matlab.

3.

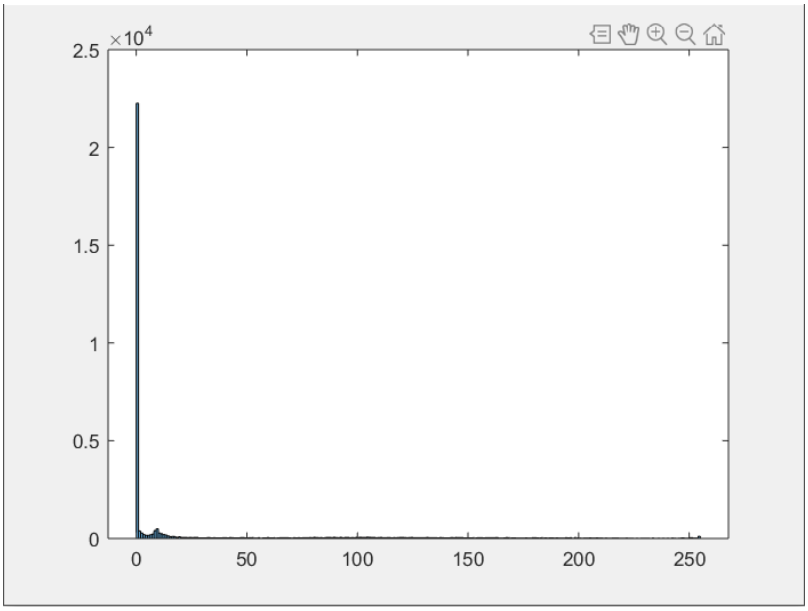
- kid.bmp

  A entropia do kid.bmp é 6.9541 bits por símbolo e o seu gráfico é:



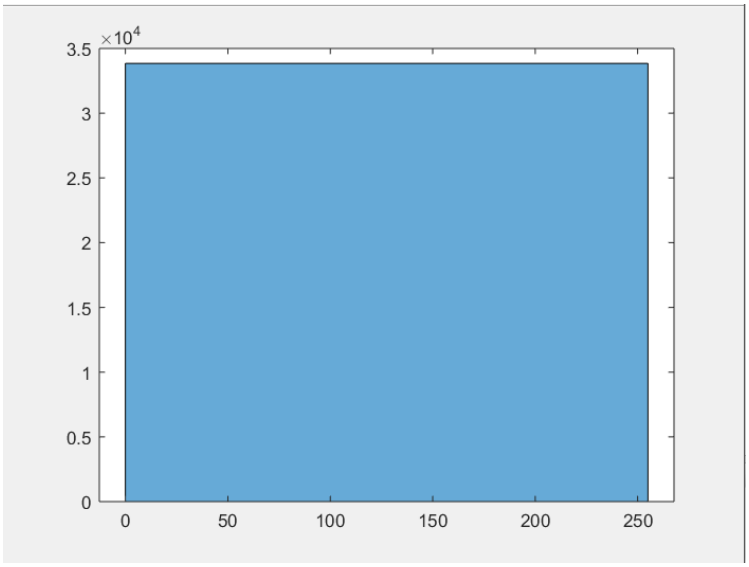
- homer.bmp

  A entropia do homer.bmp é 3.4659 bits por símbolo e o seu gráfico é:



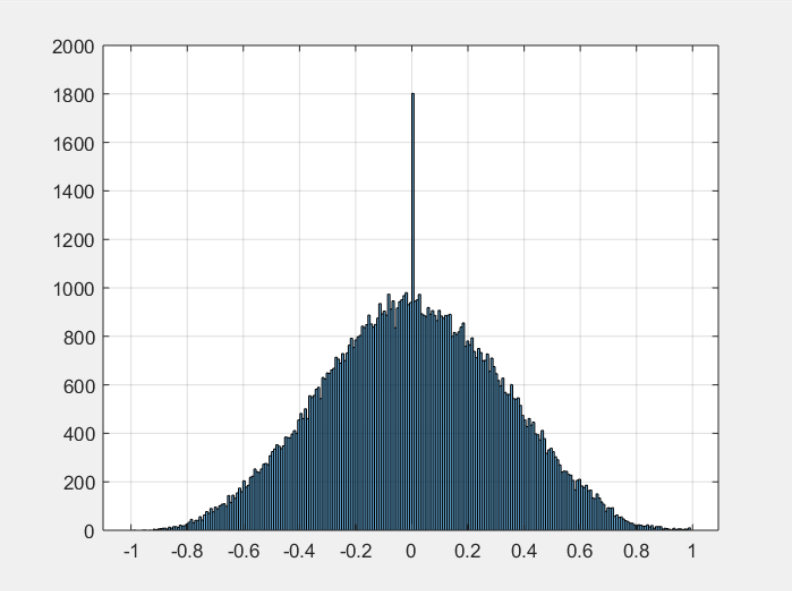
- homerBin.bmp

  A entropia do homerBin.bmp é 0.6448 bits por símbolo e o seu gráfico é:



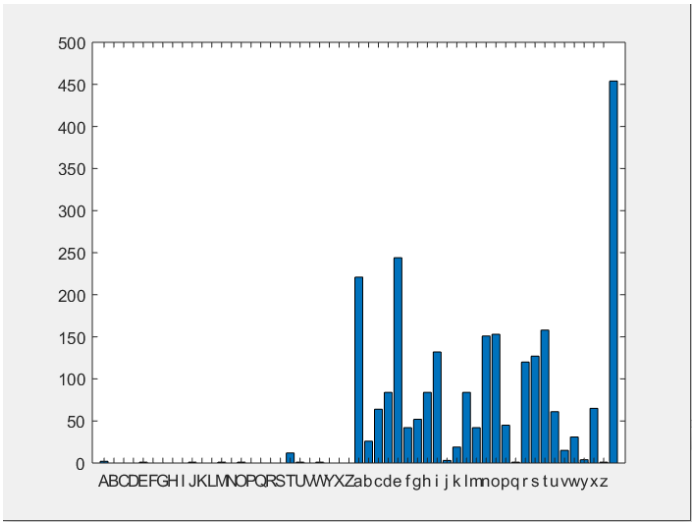
- guitarSolo.wav

  A entropia do guitarSolo.wav é 7.3580 bits por símbolo e o seu gráfico é:



- english.txt

  A entropia do english.txt é 4.2798 bits por símbolo e o seu gráfico é:



Comparando os gráficos e as entropias dos ficheiros, é possível concluir que quanto mais dispersa está a probabilidade maior é a entropia.
Podemos comprimir de forma não destrutiva, isto é, não perder símbolos, com o número de bits médio teórico correspondente ao limite mínimo teórico (entropia).

4.

Na quarta pergunta era solicitado para calcularmos a entropia utilizando as rotinas de codificação de Huffman, para isso criámos uma função, “calculaEntropiaH.m”, usando a função fornecida, “hufflen()”, enviamos como parâmetro um array com a percentagem de cada símbolo, e a função retorna o valor de bits para cada símbolo.

Calculamos a entropia com a seguinte fórmula em matlab:

```
e = sum(p(p~=0).*HLen)
```

Sendo p o array com as percentagens e Hlen o array com a quantidade de símbolos que a função hufflen() fez return.

A entropia de Huffman de kid.bmp é: 6.9832

A entropia de Huffman de homer.bmp é: 3.5483

A entropia de Huffman de homerBin.bmp é: 1

A entropia de Huffman de guitarSolo.wav é: 7.3791

A entropia de Huffman de english.txt é: 4.3033

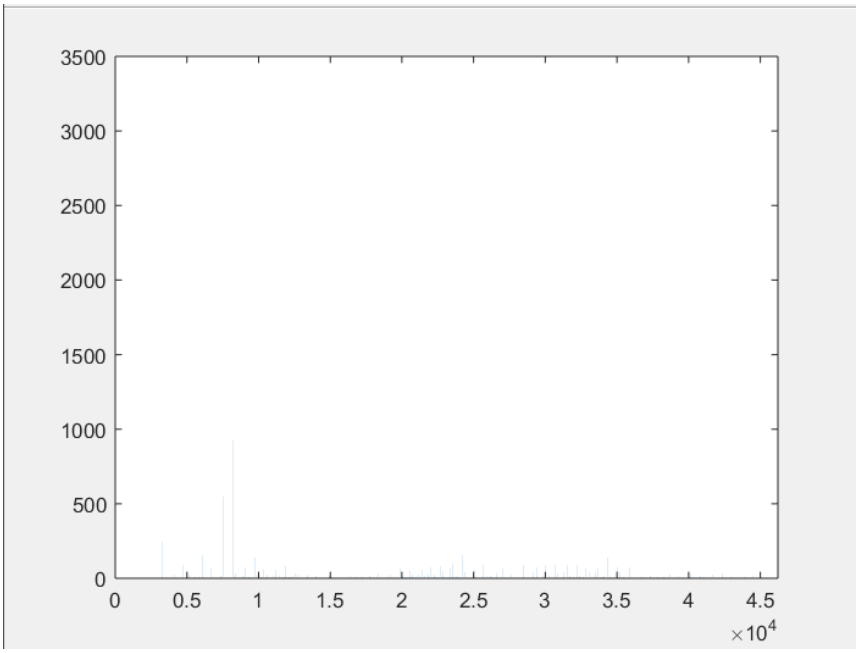
Como aprendido nas aulas, se houver uns símbolos que são usados mais vezes podemos codificá-los com menos bits e os que são menos usados com mais bits por símbolo para compensar, e assim obtemos uma média de bits por símbolo menor.

Só é possível reduzir a variância se comprimirmos de uma forma destrutiva, restringindo o alfabeto só com os elementos com maior frequência.

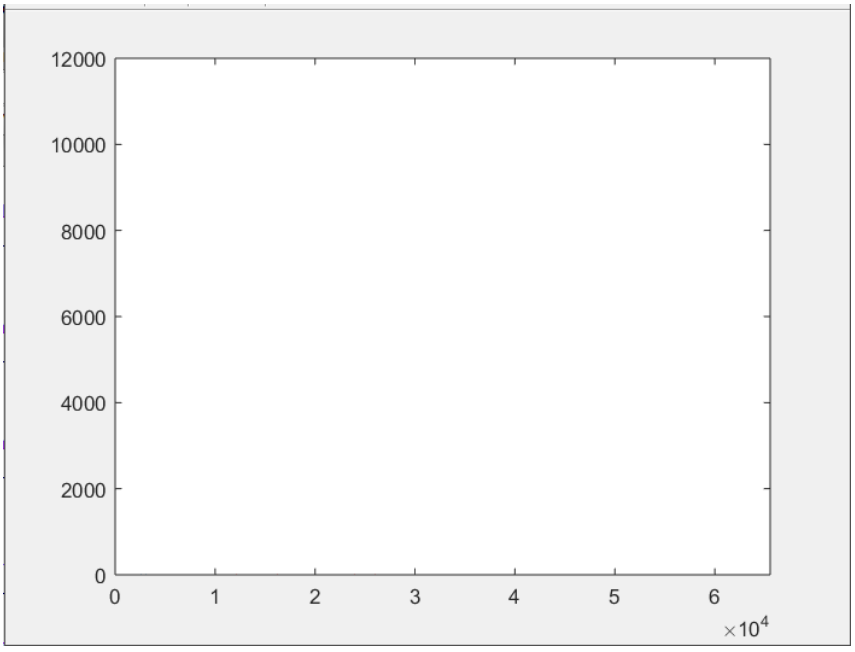
5.

Para a quinta pergunta, fizemos uma função “CalculaAgrupado.m”, que recebe a fonte como parâmetro, determina o alfabeto da fonte e cria uma matriz de zeros em que as linhas e as colunas têm o comprimento do alfabeto.
Fizemos um “for” que começa no 1 e percorre dois a dois até ao (comprimento do vetor da fonte -1), isto para não haver “segmentation fault”.
Dentro do “for”, percorre o vetor da fonte e procura o índice dos dois símbolos seguidos que vai encontrando, de seguida vai à matriz e acrescenta 1 nesses dois índices.
Assim contamos as frequências em que os dois símbolos aparecem juntos.

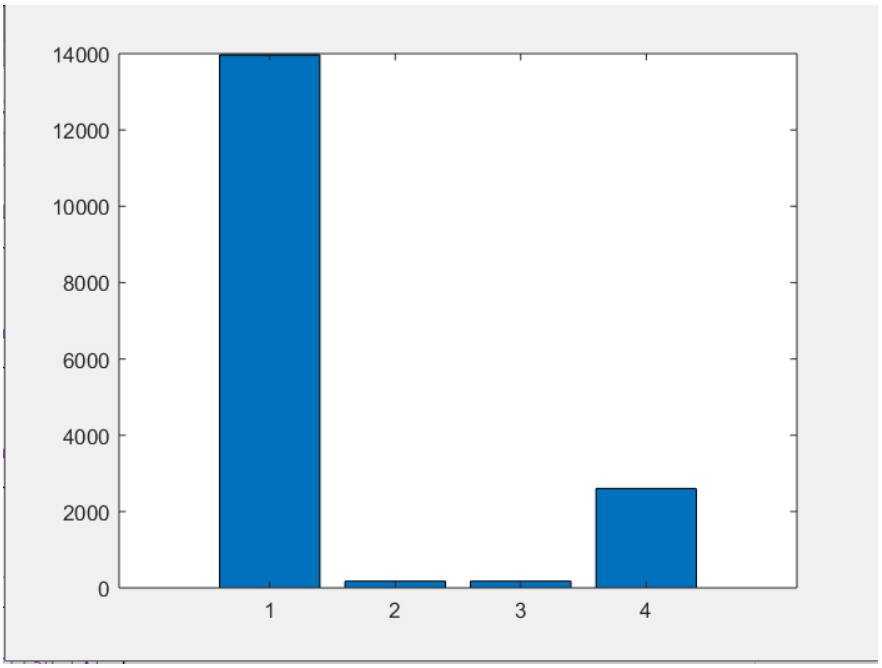
A Entropia agrupada de kid.bmp é 4.9201 e o gráfico:



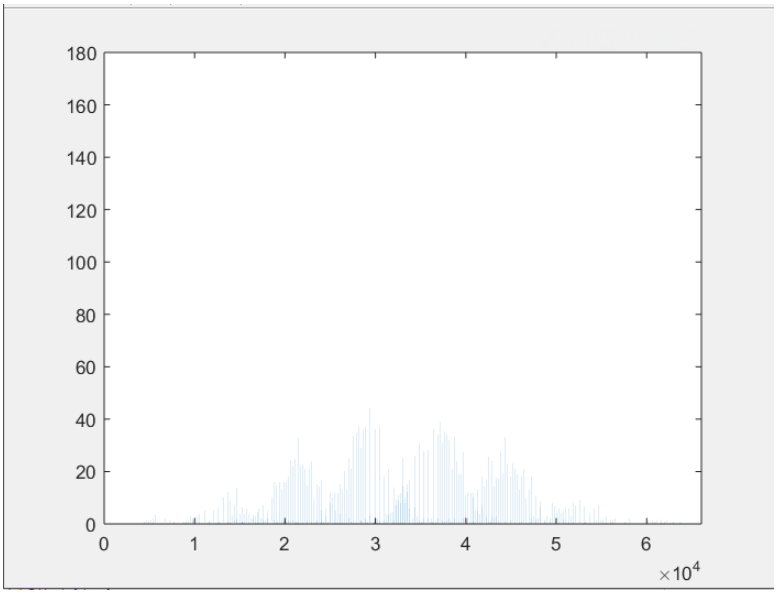
A Entropia agrupada de homer.bmp é 2.4198 e o gráfico:



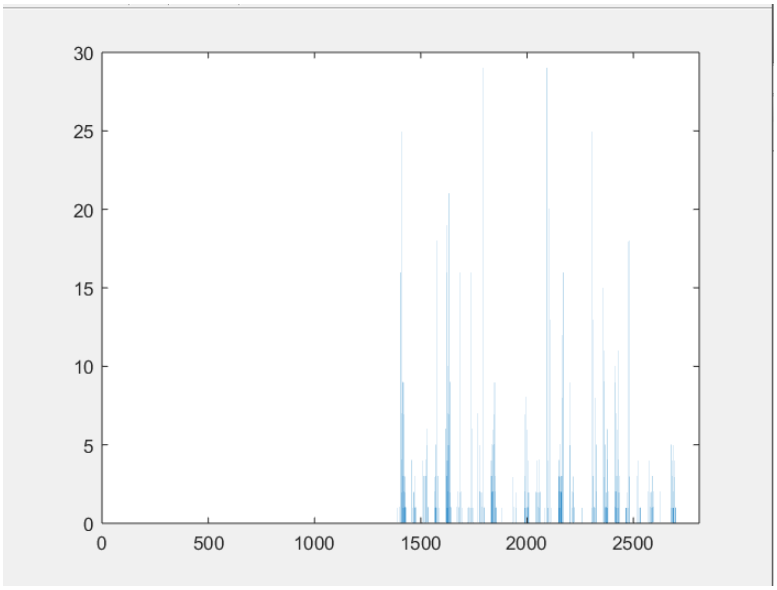
A Entropia agrupada de homerBin.bmp é 0.3907 e o gráfico:



A Entropia agrupada de guitarSolo.wav é 5.7808 e o gráfico:



A Entropia agrupada de english.txt é 3.6419 e o gráfico:



Comparando os valores da entropia agrupada com os valores de entropia calculados nas alíneas anteriores é possível concluir que a entropia é sempre menor quando é agrupada.

6.

a)

Na pergunta 6 a), calculámos a informação mútua usando a seguinte fórmula, I(X,Y) = H(X) + H(Y) – H(X,Y), determinámos a entropia de X e Y como fizemos na segunda pergunta e para a H(X,Y) usamos o mesmo raciocínio que para a entropia agrupada, mas em vez de contabilizarmos os dois símbolos seguintes, agrupamos os mesmos resultados da query e da janela deslizante.
Criamos as rotinas “CalculaMutua.m” e “EntropiaMutua.m”.

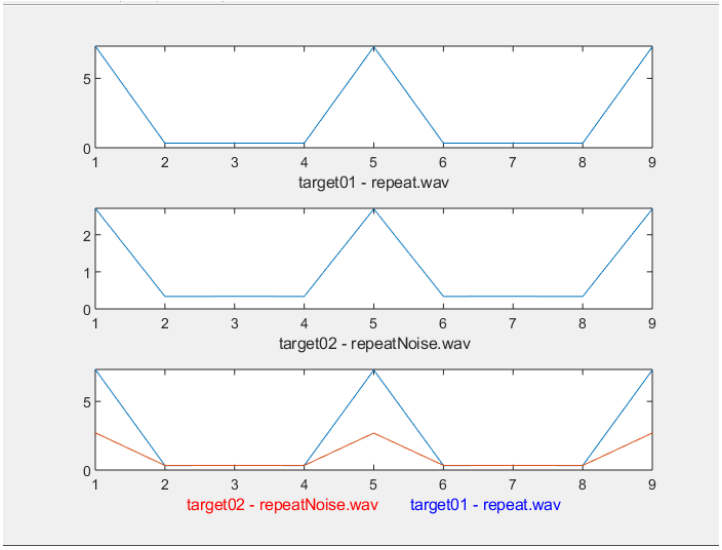
O Resultado da 6 a) foi:

Informação Mútua = [2.1219 1.9219 1.6464 2.1710 1.9710 1.7710 2.0464
2.1219 2.3219 2.5219 2.2464 2.2464 2.2464 2.2464 2.4464
2.4464 2.5219 2.7219 2.5219 2.3219 2.3219 2.1219 2.3219
2.3219 2.1219 2.0464 2.0464 2.0464 2.0464 2.0464 2.3219
2.3219 2.0464 2.1219 2.3219 1.8464 1.7710 2.0464 2.0464
2.0464 2.3219]

b)

Informação Mútua de target01 - repeat.wav:
[7.3207 0.3289 0.3354 0.3297 7.2789 0.3289 0.3355 0.3293 7.2806]

Informação Mútua de target02 - repeatNoise.wav:
[2.7113 0.3354 0.3379 0.3366 2.6934 0.3363 0.3388 0.3340 2.6946]

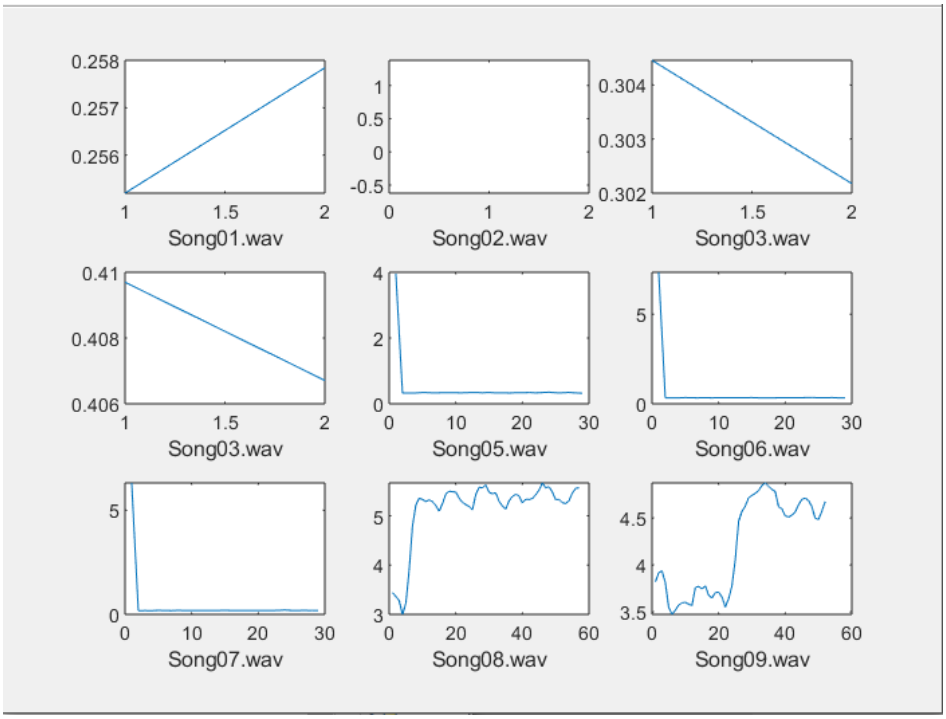


É possível analisar pelo gráfico e pelos valores que a informação é menor quando há ruído, ou seja no target 02, visto isso é possível concluir que o ruído reduz a informação mútua.

c)

Informação Mutua Máxima:

Song06.wav: 7.338379
Song07.wav: 6.313053
Song08.wav: 5.667548
Song09.wav: 4.872563
Song05.wav: 3.961751
Song04.wav: 0.409696
Song02.wav: 0.377678
Song03.wav: 0.304454
song01.wav: 0.257843



Como a finalidade deste exercício era fazer um algoritmo que identificasse o sinal sonoro mais idêntico a um ficheiro áudio numa base de dados, pela informação mútua máxima, diríamos que o som mais parecido com o ficheiro áudio era o song06, analisando os gráficos diríamos que o som mais idêntico ao ficheiro áudio na base de dados era o song08, pois é o gráfico que está mais tempo com uma informação mútua elevada.