

## Métodos Probabilísticos em Eletrotecnia (2017/18)

### Trabalho prático n° 3

#### PARTE I

Coloque no seu diretório de trabalho o ficheiro `trab03.mat` (disponível no moodle) e execute a sequência de comandos MATLAB

```
N = 512;  
load trab03.mat; % é criada a variável y  
figure(1);  
imagesc(reshape(y,N,N));  
colormap(gray);  
colorbar
```

1. Quantos valores diferentes (símbolos) tem a variável  $y$ ?
  - (a) Obtenha o histograma de  $y$ .
  - (b) De quantos bits necessita para representar os dados contidos em  $y$ ?
2. A imagem representada pela variável  $y$  deve ser transmitida por um canal ruidoso que introduz ruído aditivo Gaussiano de média nula e desvio padrão sigma, ou seja as amostras do sinal recebido  $y_{rx,n}$   $n=1,2,\dots$ , vem dadas por:

$$y_{rx,n} = y_{tx,n} + w_n, n=1,2,\dots$$

Onde  $y_{tx,n}$   $n=1,2,\dots$ , são as amostras à saída do emissor e  $w_n$  as amostras de ruído que se somam ao sinal transmitido.

- (a) Esboce um programa que:
  - Retire a componente DC ou seja que centre a variável  $y$  em 0 (numa transmissão não se deve desperdiçar potência enviando sinais DC).
  - Some ruído Gaussiano com média nula e desvio padrão sigma, o que pode ser feito usando a função `randn`.
  - Regenere os símbolos transmitidos i.e. com base na observação do símbolo corrompido por ruído escolha o mais próximo (pode usar função `round`)
  - Calcule a relação entre potência do sinal transmitido e potência do ruído (SNR)
  - Reconstrua e mostre imagem, para o que pode usar o comando  
`imagesc(reshape(sinal_regenerado,N,N))`
- (b) Corra o programa para uns 4-5 valores de SNR entre 100 e 1 (20dB a 0dB) e relacione a qualidade da imagem reconstruída com SNR.

3. Converta para binário a variável  $y$  (e no recetor tem de converter novamente para decimal) e repita 2 (b).

Neste caso os dados binários deverão ser polares (-1 e 1) o que dá potência de 1 e consequentemente valores do ruído tem de ser ajustados.

Para converter uma sequência decimal para binária pode usar o seguinte código

```
% n n° bits usado na representação de elemento de y
%dec2bin converte cada elemento de y num string de 0 e 1
binary_string=dec2bin(y,8)

binary_data=rem(double(binary_string),2); % converte string em n°s
balanced_binary_data=2*binary_data-1; % conversão para polar
(outra opção é usar função de2bi que converte directamente para valores numéricos se
essa função estiver incluída na versão do Matlab que usa)
```

Para converter uma sequência de 8 bits para decimal pode fazer

```
decimal_value=binary_byte*[128 64 32 16 8 4 2 1]' )
% binary_byte vetor linha neste caso
```

4. Considere uma nova variável  $y_{dif} = \text{diff}(y)$ .
  - (a) Faça o histograma de  $y_{dif}$  e compare com o histograma de  $y$ . Explique, qualitativamente, as diferenças.
  - (b) Com base nos histogramas indique se poderia ganhar alguma coisa em termos da quantidade de bits transmitidos usando  $y_{dif}$  em vez de  $y$
5.
  - (a) Quantifique as amostras de  $y$  em 7 bits ( $7 < n$ ) criando nova variável  $y_q$ . Retire simplesmente os  $n-7$  bits menos significativos.
  - (b) Crie a variável  $y_{difq} = \text{diff}(y_q)$
  - (c) Codifique  $y_{difq}$  para binário e transmita (seguindo os mesmos procedimentos de 3)
  - (d) Mostre que para reconstruir a sequência  $y_q$  pode usar um filtro IIR  $H(z) = 1/(1 - z^{-1})$
  - (e) Reconstrua a sequência  $y_q$  usando função `filter` e observe a imagem resultante na ausência de ruído e com ruído igual aos valores usados em 3.
  - (f) Repita os pontos (a),..., (e) para valores quantificados em 6, 5 e 4 bits.
  - (g) Deverá observar que a distorção na imagem apresenta um aspecto que ao invés do caso 3 parece ser mais estruturada. Como explica isso?

## PARTE II

No exercício anterior as variáveis  $y$ ,  $y_q$ ,  $y_{dif}$  e  $y_{difq}$  podem ser consideradas variáveis aleatórias. Considerando que precisa de oito símbolos para as variáveis  $y_q$  e  $y_{difq}$ ,

1. Calcule as pmf de  $y_q$  e  $y_{difq}$  para uma quantificação de 7 bits
2. Há alguma relação entre a pmf e o histograma?
3. Pretende-se armazenar a variável  $y_{difq}$  num ficheiro binário. Para isso ordenaram-se os diferentes símbolos de  $y_{difq}$  por ordem decrescente de probabilidade e atribuiu-se a cada símbolo um dos códigos da tabela

Prob.	Codigo A	Codigo B	Codigo C
1	000	0	11111110
2	001	10	1111110
3	010	110	111110
4	011	1110	11110
5	100	11110	1110
6	101	111110	110
7	110	1111110	10
8	111	11111110	0

- (a) Calcule o número de bits requerido por cada código para guardar  $y_{difq}$ .

- (b) Calcule o número medio de bits por símbolo usado por cada código.
  - (c) Indique qual é o código mais eficiente, ou seja, aquele que permite guardar a informação em menos bits.
4. Repita o procedimento anterior para a variável  $y_q$  e comente os resultados.