



## LISTA DE EXERCÍCIOS (10/10) – Entregar em 05/6/2019

### 1-Simulação de Sinais Estocásticos com $N=2^{12}$ valores de medidas.

1.1. Utilize o algoritmo *powennoise.m* e gere os seguintes ruídos  $1/f^\beta$ :

S1:  $\beta=0$  (white noise)

S2:  $\beta=1$  (pink noise)

S3:  $\beta=2$  (red noise)

1.2. S4: Série Caótica a partir do Mapeamento Quadrático (Logístico) para  $p=3.85$ , considerando  $A_0=0.001$ .

1.3. Gere o algoritmo em python para somar e normalizar com  $\langle A \rangle = 0$ , dois sinais com o mesmo tamanho e construa a série  $S_5=S_1+S_4$ ,  $S_6=S_2+S_4$  e  $S_7=S_3+S_4$ .

1.4. Utilize o algoritmo *pmodel.m* e gere os seguintes sinais com  $N=2^{12}$  valores de medidas:

S8:  $p=0.52$ ,  $\beta=-1.66$

S9:  $p=0.62$ ,  $\beta=-0.45$

S10:  $p=0.72$ ,  $\beta=-0.75$

<http://www2.meteo.uni-bonn.de/staff/venema/themes/surrogates/pmodel/pmodel.m>

(Bônus de 5 pontos na lista para quem entregar a versão *pmodel.py*)

### 2- Distribuições de Probabilidades

2.1. Escreva um programa em python que, para uma amostra com  $N$  resultados, ajuste as seguintes PDFs:

(i) Uniforme, (ii) Binomial, (iii) Beta, (iv) Laplace, (v) Gamma, (vi) Exponencial  
(v) Qui-quadrado, (vi) Cauchy, (vii) Beta e (viii) Gaussiana (*Normal*)

2.2. Considere o seguinte experimento: Lançamento de 3 dados simultâneos com registro de quantas vezes um determinado resultado pode ser obtido.

Mostre que a distribuição limite é binomial e que com  $N$  tendendo a infinito ela converge para uma Gaussiana.

### 3-Probabilidade Condicional

3.1. Considere 3 regiões do céu contendo aproximadamente o mesmo número (N) de galáxias, cujas Distribuições morfológicas dada por um modelo seja aquela apresentada na Tabela abaixo.

Tipo \ Região	S1	S2	S3
Irregulares	10%	25%	15%
Espirais	60%	40%	55%
Elípticas	30%	35%	30%

Em cada região será realizado um survey (S1, S2 e S3) considerando para cada uma um telescópio. Supondo que os telescópios são equivalentes e que as observações serão aleatórias calcule as seguintes probabilidades:

- i) A primeira galáxia observada ser espiral ou elíptica.
- ii) Se a primeira galáxia observada for irregular, qual a probabilidade dela pertencer à região do survey S1.

3.2. Considere o exercício anterior e crie um “bootstrap” para gerar 10 amostras contendo 200 galáxias cada uma. Considere os valores de morfologia caracterizados pelo parâmetro g1 (da técnica gradient pattern analysis) dado na tabela abaixo.

Irregulares: 1.97-1.99  
Espirais: 1.96-1.98  
Elípticas: 1.92-1.96

Aplice o Teorema de Bayes para encontrar a máxima verossimilhança considerando os modelos Gaussiano.

3.3. Supondo que SNla ... ..estime tempo de telescópio....

### 4- Teorema do Valor Extremo

Reconsidere o exercício anterior e aplique o Teorema de Bayes para encontrar a máxima verossimilhança considerando o modelo GEV.

### 5- Classificação de Cullen-Frey

Classifique a população de amostras geradas no exercício 3.2. no espaço de Cullen-Frey, calcule os desvios e compare os desempenhos dos modelos.

## 6- PSD & DFA: S3, S7, S8.

6.1. Considere as séries temporais listadas acima e obtenha os valores dos respectivos índices espectrais:  $\beta$  (via PSD) e  $\alpha$  (via DFA).

6.2. Confira se o PSD está bem ajustado a partir da formula WKP:  $\beta = 2\alpha - 1$ .

6.3. Repita 6.1. para (a) ST-Sol3GHz, (b) ST-surftemp504 e (c) uma ST de sua escolha (alternativa).

6.4. Com base nos valores de  $\beta$  (obtido via PSD e calculado via formula WKP).

## 7- Singularity Multifractal Spectra (SMS), também conhecido como MDFA.

7.1. Considere o programa `mfdfa.py`. Aprimore o programa para o mesmo calcule o índice  $\Psi = \Delta\alpha / \alpha_{\max}$ .

7.2. Obtenha o espectro de singularidade para todos os sinais do exercício 6.

7.3. Com base nos valores obtidos em 7.2., discuta os possíveis processos subjacentes para cada ST.

8- Considere o software SPECTRUM e discuta, de forma comparativa e sucinta, outros métodos para análise espectral de sinais não abordados no curso.

## 9- Global Wavelet Spectrum

9.1. Utilize o `Waipy` para obter o GWS (Morlet) de todos as ST do exercício 6.1.

9.2. Repita 9.1. utilizando uma Db8.

## 10- Self-Organized Criticality (SOC)

i) Calcule a Taxa Local de Flutuação  $[Y_i]$  para cada valor da ST

ii) Calcule  $P[Y_i] = \text{counts}(n_i) / N$

iii) Plot  $\log P[Y_i] \times \log n_i$  (e ajuste uma lei de potencia).

10.1. Implemente um algoritmo em Python para caracterização de SOC a partir de uma ST.

10.2. Aplique o SOC.py para todas as ST do exercício 6.1.

## 11- EXERCÍCIO BONUS: +1.0 NA NOTA FINAL

Crie uma suíte em Python que tem como entrada uma ST e como saída um classificador do padrão de variabilidade da ST em um plano  $\beta^* \times \Psi$