Funções Úteis

Erick Amorim Fernandes 86301
ELT 451 - Inteligência Computacional
Departamento de Engenharia Elétrica, Universidade Federal de Viçosa, Viçosa - MG
erick.fernandes@ufv.com

***Resumo*—Este trabalho irá demonstrar algumas ferramentas básicas utilizadas na disciplina de Inteligência Computacional (ELT 451). Para isso, será criada uma população de pontos com distribuição gaussiana e outra com distribuição uniforme. Por fim, será criada uma função que meça a distorção entre duas séries temporais.**

***Palavras Chaves*—Inteligência Computacional, Distribuição Gaussiana, Distribuição Normal, Distorção entre séries temporais.**

I. Introdução

Para a criação de modelos computacionais inteligentes faz-se necessário o uso de técnicas que implementem tomadas de decisões a computadores, além de bancos de dados para treinar e avaliar a acurácia de tais modelos. [1]

Uma forma eficiente de se criar dados para testes iniciais é gerar populações com distribuição previamente selecionadas e analisar a resposta do modelo após seu processamento.

Dessa forma, visando o primeiro contato do aluno com o curso de Inteligência Computacional (ELT451), serão abordados dois modelos de distribuição para a criação de populações: Gaussiana e Uniforme.

A distribuição gaussiana, também conhecida como distribuição normal, é considerada como uma das mais importantes dentre as distribuições estatísticas. Possui como marca registrada valores iguais para moda, média e mediana, seu formato de sino, sua simetria e sua larga aplicação em determinações probabilísticas. [2]

Por outro lado, a distribuição uniforme ocorre quando valores finitos dentro de um intervalo possuem chances iguais de ocorrer. [3]

Outra habilidade importante para o estudo de inteligência computacional é a capacidade de realizar comparações entre saídas, uma forma elegante de introduzir esse conceito será abordado através da criação de uma função que meça a distorção (em dB) entre duas séries temporais.

II. Objetivo

Gerar populações com distribuições gaussiana e uniforme utilizando as funções *"randn"* e *"rand"* disponíveis no *software Matlab*. Por fim, medir a distorção entre duas séries temporais, lançando mão da função *"log10"*.

III. Materiais e Métodos

A distribuição normal pode ser definida por dois parâmetros, a média (μ), centro da dispersão, e a variância ($\sigma^2 > 0$) que descreve o seu grau de dispersão. De tal modo, pode-se dizer que X possui uma distribuição normal e aproximadamente igual a $N(\mu;\sigma^2)$, representada pela figura 1.

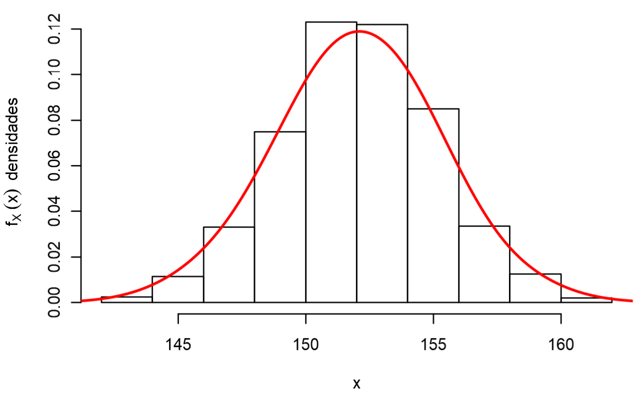

Fig. 1. Curva característica de uma função normal.

Sendo a distribuição reduzida padronizada (Z), dada por 1, onde X é uma variável aleatória normal com média ($\mu, \sigma^2 > 0$).

$$Z = \frac{X-\mu}{\sigma} \tag{1}$$

No *software Matlab* a função responsável pela geração de um vetor aleatório de distribuição gaussiana é descrita por *"randn"*. Desse modo, a criação de uma classe no plano para tal distribuição, pode ter seu deslocamento dado pela soma da média desejada, assim como a dispersão dos dados através da multiplicação pelo desvio padrão.

Para o cálculo de distorção foi usada a equação 2.

$$d_{xy} = 10\log\left(\frac{E[(x(k)) - y(k))^2]}{E[x^2(k)]}\right) \tag{2}$$

IV. Resultados

Aplicando a formulação anteriormente apresentada são obtidas as figuras 2 e 3.

Para gerar a população por distribuição normal, figura 2, foi escolhido uma média de dez para a abscissa e vinte para a ordenada, uma variância igual a um para os dois eixos e um total de dois mil pontos para serem distribuídos.

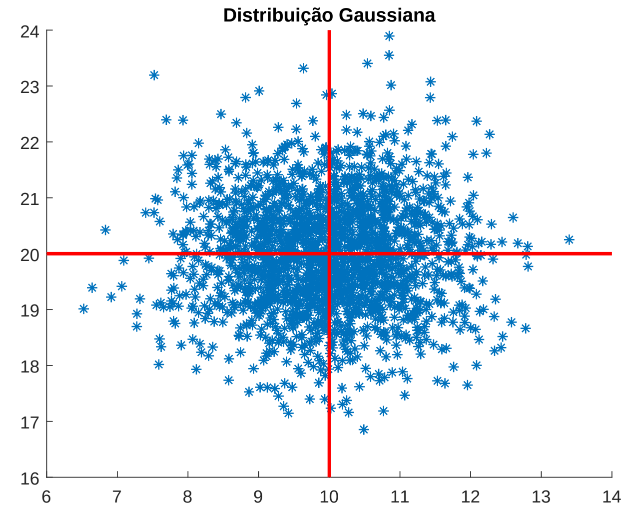

Fig. 2. População de 2000 pontos em distribuição normal.

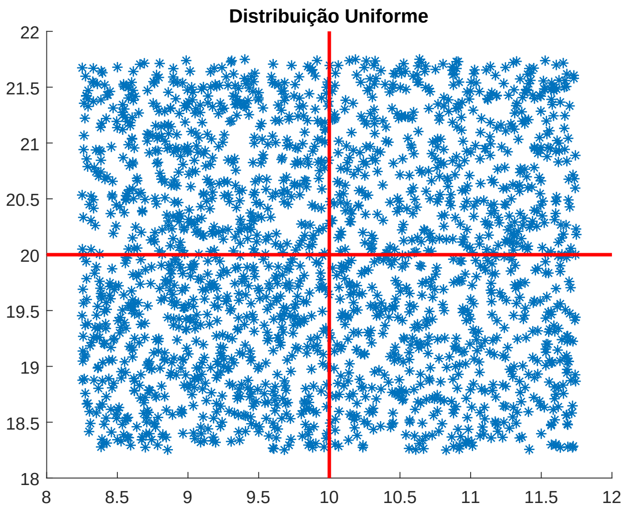

Fig. 3. População de 2000 pontos em distribuição uniforme.

É possível observar na figura 2 que a distribuição dos pontos se concentram em torno das médias estabelecidas, enquanto a dispersão se relaciona diretamente com a variância dada ao ponto.

Já para a distribuição uniforme, figura 3, foi criada uma população de 2000 pontos a partir da função *rand*, ou seja por distribuição normal.

Quando comparada com a figura 2 fica evidente a diferença entre as duas metodologias de distribuição, enquanto a primeira possui seus pontos concentrados próximo ao seu centro, a segunda possui seus pontos distribuídos de maneira uniforme dentro dos limites pré-estabelecidos.

Para o estudo de distorção de séries temporais foi usado como entrada as funções senoidais representadas pela figura 4 na equação 2 e obteve-se um resultado aproximadamente igual a 9,54 Db.

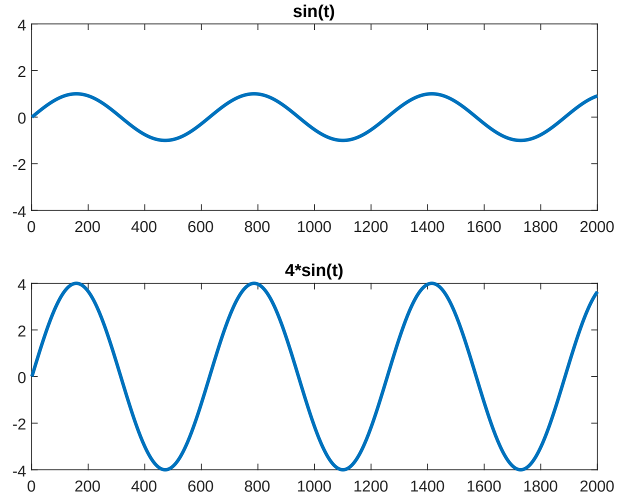

Fig. 4. Erro linear da formação nos eixos x e y.

V. Conclusão

Este relatório demonstrou as principais características de uma população em distribuição uniforme e gaussiana, foi possível, também, após o processamento via *matlab*, visualizar as diferenças entre os dois tipos de distribuição, enquanto uma possui seus pontos concentrados próximos as suas médias o outro possui seus pontos distribuídos por todo o espaço definido.

Por fim, foi calculado a distorção entre duas senoides.

Além do mais, para a resolução dos problemas fez-se necessário o aprendizado de ferramentas computacionais e de conceitos matemáticos que servirão para estudos futuros relacionados a disciplina de inteligência computacional.

Referências

[1] I. N. da Silva, D. H. Spatti, and R. A. Flauzino, *Redes Neurais Artificiais para engenharia e ciências aplicadas*. Aetliber, 2016, vol. 2.

[2] "O que é, para que serve e como calcular a distribuição normal?" https://www.voitto.com.br/blog/artigo/distribuicao-normal, accessed: 2021-08-15.

[3] "Distribuição de probabilidades – distribuições uniforme, log-normal, binomial e t de student," https://proeducacional.com/ead/curso-cga-modulo-i/capitulos/capitulo-4/aulas/distribuicao-de-probabilidades-distribuicoes-uniforme-log-normal-binomial-e-t-de-student/, accessed: 2021-08-15.

```matlab
%% Ditribuição Gaussiana

%% Entradas

mx = 10 ; % média de x
my = 20 ; % média de y
vx = 1; % variancia de x
vy = 1; % variancia de y

%% População

X = randn(2000,1);
Y = randn(2000,1);

%% Tratamento para distribuição

xg = (sqrt(vx).*X+mx); % Aplicação da Gaussiana em x
yg = (sqrt(vy).*Y+my); % Aplicação da Gaussiana em y

%% Criar linhas de referência

x1 = [max(xg),min(xg)];
y1 = [max(yg),min(yg)];

v1 = [mx,mx];
v2 = [floor(y1(2)),ceil(y1(1))];

h1 = [floor(x1(2)),ceil(x1(1))];
h2 = [my,my];

%% Plotagem

figure()
title ('Distribuição Gaussiana');
hold on
plot(xg, yg, '*');
plot(h1, h2, 'r' ,'LineWidth',2);
plot(v1, v2, 'r' ,'LineWidth',2);
axis([h1(1) h1(2) v2(1) v2(2)]);

%% Distribuição Uniforme com desvio

%%

Xu = rand(2000,1);
Yu = rand(2000,1);
```

```
%%

amin = (2*mx - sqrt(12*vx))/2;
amax = (2*mx + sqrt(12*vx))/2;
bmin = (2*my - sqrt(12*vy))/2;
bmax = (2*my + sqrt(12*vy))/2;

xu = amin + (amax - amin).*Xu;
yu = bmin + (bmax - bmin).*Yu;

%% Criar linhas de referência

x1u = [max(xu),min(xu)];
y1u = [max(yu),min(yu)];

v1u = [mx,mx];
v2u = [floor(y1u(2)),ceil(y1u(1))];

h1u = [floor(x1u(2)),ceil(x1u(1))];
h2u = [my,my];

%% Distribuição Uniforme sem desvio


x2= mx-3.5*vx/2+ 3.5*rand(1,2000)*vx;
y2= my-3.5*vy/2+ 3.5*vy*rand(1,2000);

%% Plotagem

figure()
title ('Distribuição Uniforme');
hold on
% plot(xu, yu, '*');
plot(x2, y2, '*');
plot(h1u, h2u, 'r' ,'LineWidth',2);
plot(v1u, v2u, 'r' ,'LineWidth',2);
axis([h1u(1) h1u(2) v2u(1) v2u(2)]);



%% Parte 2 - Distorção


clear all
close all
clc
```

```matlab
%% Criação de duas séries temporais

% x1 = timeseries(rand(10,1));
% x2 = timeseries(rand(10,1));

t = [0:0.01:20];
x1 = sin(t);
x2 = 4*sin(t);

subplot(2,1,1)
plot(x1,'LineWidth',2)
axis([0,2000,-4,4])
title ('sin(t)');
subplot(2,1,2)
plot(x2,'LineWidth',2)
axis([0,2000,-4,4])
title ('4*sin(t)');

%% Adicionar dados da série e uma variável

% S1 = x1.data;
% S2 = x2.data;

S1 = x1;
S2 = x2;

%% Aplicando a fórmula para cálculo de distorção

soma_num = 0;
soma_den = 0;

for i = 1 : length (S1)

    num = (S1(i)-S2(i)).^2;
    soma_num = soma_num + num;

    den = (S1(i)).^2;
    soma_den = soma_den + den;

end

media_num = soma_num/ length(S1);
media_den = soma_den/ length(S1);

dxy = 10*log10(soma_num/soma_den)
```