Funções Úteis

Erick Amorim Fernandes 86301
ELT 451 - Inteligência Computacional
Departamento de Engenharia Elétrica, Universidade Federal de Viçosa, Viçosa - MG
E-mail: 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.

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. **Procurar algumar ref**

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. **Referencia1**

Por outro lado, a distribuição uniforme ocorre quando valores finitos dentro de um intervalo possuem chances iguais de ocorrer. **Referencia2**

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. OBJETIVOS

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 gaussiana

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

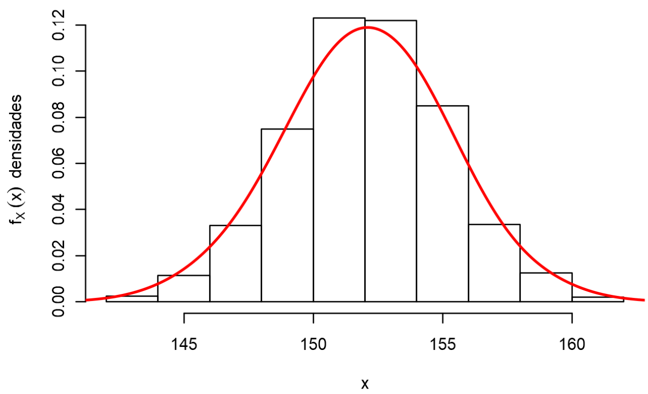

Figura 1 - Função de densidade de probabilidade para a variável aleatória contínua

Sendo a distribuição reduzida padronizada (Z), dada por (1), onde X é uma variável aleatória normal com média (μ, σ2 > 0).

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

No *software* Matlab a função responsável pela geração de um vetor aleatório de distribuição gaussiana é dada 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).

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

IV. RESULTADOS E DISCUSSÕES

Tendo em vista a metodologia explicada no tópico anterior, os resultados encontrados para tais simulações encontram-se dispostos a seguir. A Figura 4 mostra a classe gerada no plano através de uma distribuição gaussiana obtida pelo emprego da função *randn.* Nesse caso, com os seguintes parâmetros de entrada média dez para abcissas, vinte para ordenadas e variância um para ambas.

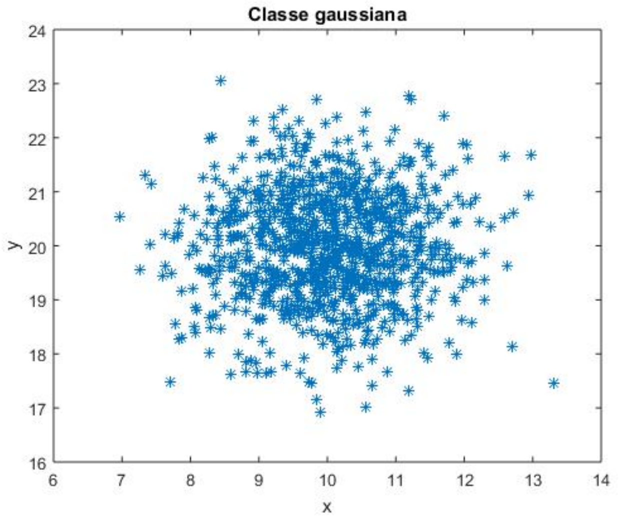

Figura 4 - Classe gerada por distribuição gaussiana no plano.

É importante salientar que, pela observação da Figura 4, a distribuição dos pontos concentra-se em torno das médias estabelecidas pelo usuário para cada eixo, enquanto a maior ou menor dispersão está relacionada a variância informada.

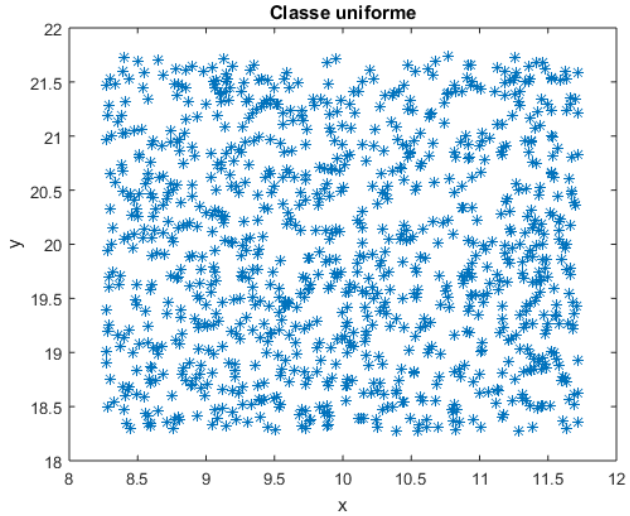

Figura 5 - Classe gerada por distribuição uniforme no plano.

A Figura 5 ilustra a classe gerada através da função *rand,* isto é, pela distribuição uniforme. Assim, é perceptível a diferença entre a distribuição dos dados quando comparada à distribuição anterior, uma vez que, nesta última, os pontos são dispostos por todo intervalo.

Em relação ao cálculo da distorção entre séries temporais, empregou-se vetores gerados por distribuição aleatória, assim os valores d_{xy} encontrados diferem consideravelmente a cada tentativa de execução do código. Na Tabela 1 a seguir estão dispostos os dados de duas séries temporais, a partir das quais realizou-se o cálculo de distorção.

Tabela 1 - Dados das séries temporais utilizadas

t	Dados 1	Dados 2
0	0.3181	0.3463
1	0.9028	0.9291
2	0.2230	0.5083
3	0.9571	0.2083
4	0.2397	0.5476
5	0.1770	0.7960
6	0.5108	0.3666
7	0.8377	0.8127
8	0.8851	0.1078
9	0.9874	0.8674

Para os valores indicados na Tabela 1, a distorção encontrada entre as séries foi de -21.0519.

V. CONCLUSÕES

Assim, a realização da prática permitiu empregar algumas funções de grande utilidades em pesquisas, de modo a analisar características particulares de cada uma delas, além disso, foi possível desenvolver novas funções que apresentam significativa relevância para futuras aplicações, possibilitando , por exemplo, a geração de gráficos e o cálculo de distorções de forma mais eficiente. Desse modo, os objetivos estabelecidos inicialmente foram alcançados com êxito.

REFERÊNCIAS BIBLIOGRÁFICAS

[1] RODRIGUES, Erica Castilho, Variáveis Aleatórias Discretas e Distribuição de Probabilidades - parte II, Cap. 3, 2016. Disponível em :< http://professor.ufop.br/sites/default/files/ericarodrigues/files/cap3parte2.pdf> Acesso em: 8 de março de 2020.

[2] REIS, Marcelo Menezes, Análise de séries temporais, UFSC. Cap.4. Disponível em :<https://www.inf.ufsc.br/~marcelo.menezes.reis/Cap4.pdf> Acesso em: 8 de março de 2020.

APÊNDICE A

```
%% Exercício 1
%Distribuição gaussiana
 function [xgauss ygauss]=clasgauss(mx,vx,my,vy)
% Exibe a classe gaussiana no plano, uma vez fornecidas a media e
% variância dos dados.
X=randn(1000,1);
Y=randn(1000,1);
xgauss=(sqrt(vx).*X+mx);
ygauss=(sqrt(vy).*Y+my);
figure()
plot(xgauss,ygauss, '*');
title('Classe gaussiana');
xlabel ('x'), ylabel('y');
end

%Distribuição uniforme
function [xuni yuni]=clasuni(mx,vx,my,vy)
% Exibe a classe uniforme no plano, uma vez fornecidos os limites
% inferior e superior do intervalo de cada eixo.
X=rand(1000,1);
Y=rand(1000,1);
xi=(2*mx-sqrt(12*vx))/2; %intervalo inferior das abcissas
xs=(2*mx+sqrt(12*vx))/2; %intervalo superior das abcissas
yi=(2*my-sqrt(12*vy))/2; %intervalo inferior das ordenadas
ys=(2*my+sqrt(12*vy))/2; %intervalo superior das ordenadas
xuni=xi+(xs-xi).*X;
yuni=yi+(ys-yi).*Y;
figure()
plot(xuni,yuni, '*')
title('Classe uniforme');
xlabel ('x'), ylabel('y');
end

%% Exercício 2
% Distorção entre séries temporais
function dxy= deltatemp(x,y)
%Calcula a distorção (em dB) entre duas séries temporais dadas
t=0;
g=0;
d1=x.data
d2=y.data
for k=1: length (x)
    A=(d1(k)-d2(k)).^2;
    t=t+A;
    B=(d1(k)).^2;
    g=g+B;
end
e1=t/length(x);
e2=g/length(x);
dxy=10*log10(e1/e2)
end
```