**Problemas em Equipe 06**

Estudantes:

Eduardo Eiji Goto; Gustavo Hammerschmidt; João Vitor Andrioli de Souza.

**1 – Distribuições Multivariadas Contínuas**

1. Gráfícos 3D de funções de duas variáveis

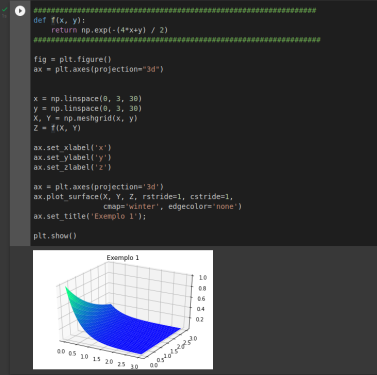
O notebook Jupyter Plot3D.ipynb contém um mini tutorial para plotar gráficos em 3D com o MatPlotLib. Você aprenderá como as funções mostradas em sala de aula são plotadas em 3D, incluído a Gaussiana Bivariada Padrão. Você também pode usar os arquivos Plot3D.py e GaussianaBivariadaPadrao.py.

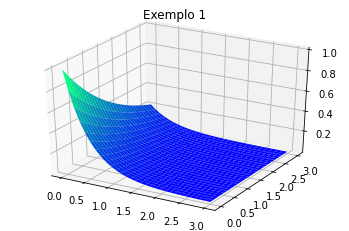
1. A função de densidade conjunta de duas variáveis aleatórias *X* e *Y* é dada por:



1. Plotar o gráfico de *fX*,*Y*(*x*, *y*)

Veja as dicas em Plot3D.ipynb e copiar o gráfico aqui





1. Encontrar *fX*(*x*) e *fY*(*y*)

Para encontrar *fX*(*x*) integrar *fX*,*Y*(*x*, *y*) em *y*

(usar o comando abixo no Wolfram e copiar o resultado)

integrate[exp[-(4\*x+y)/2]] {y from 0 to infinity}

integral_0^∞ exp(-1/2 (4 x + y)) dy = 2 e^(-2 x)

## Series expansion of the integral at x=0

2 - 4 x + 4 x^2 - (8 x^3)/3 + (4 x^4)/3 + O(x^5)
(Taylor series)

integral exp(-1/2 (4 x + y)) dy = -2 e^(-2 x - y/2) + constant

Para encontrar *fY*(*y*) integrar *fX*,*Y*(*x*, *y*) em *x*

(usar o comando abixo no Wolfram e copiar o resultado)

integrate[exp[-(4\*x+y)/2]] {x from 0 to infinity}

integral_0^∞ exp(-1/2 (4 x + y)) dx = e^(-y/2)/2

Series expansion of the integral at y=0

1/2 - y/4 + y^2/16 - y^3/96 + y^4/768 + O(y^5)
(Taylor series)

integral exp(-1/2 (4 x + y)) dx = -1/2 e^(-2 x - y/2) + constant

1. Calcular *E*[*X*,*Y*]

(usar o comando abaixo no Wolfram e copiar o resultado)

integrate[x\*2\*exp(-2\*x)] {x from 0 to infinity}

integral_0^∞ x 2 exp(-2 x) dx = 1/2 = 0.5

integral x 2 exp(-2 x) dx = 2 e^(-2 x) (-x/2 - 1/4) + constant

(usar o comando abaixo no Wolfram e copiar o resultado)

integrate[y\*1/2\*exp(-y/2)] {y from 0 to infinity}

integral_0^∞ 1/2 y exp(-y/2) dy = 2

integral1/2 y exp(-y/2) dy = 1/2 e^(-y/2) (-2 y - 4) + constant

1. Verificar se as duas variáveis são independentes.



Portanto as variáveis são independentes.

Exercício já está resolvido. Apenas para reforçar o método de veificação de independência de distribuições contínuas.

**2 – Detecção de Outliers**

Usar um modelo baseado na Gaussiana Multivariada para detectar observações anômalas. O exercício é dividido em duas partes. Na primeira, ajustar a distribuição Gaussiana Multivariada para um dataset com duas características (bidimensional), o que permite visualizar os dados e os resultados, para então encontrar valores com baixa probabilidade que serão classificados como anomalias. Na segunda parte, aplicar o algoritmo de detecção de anomalia para um dataset com maior número de dimensões.

Utilizar o Livescript DetAnomalia.

1. Modificar o arquvo estimarGaussiana.m e copiar o código modificado aqui.

function [mu, sigma] = estimarGaussiana(X)

n = size(X, 2);

mu = mean(X);

sigma = cov(X);

end

Obs.: no arquivo .mlx, eu modifiquei a chamada da função para: estimarGaussiana(...); sem o Resposta no nome.

1. Modificar o arquivo selecionarLimiar.m e copiar o código modificado aqui.

function [melhorEpsilon, melhorF1] = selecionarLimiar(yval, pval)

melhorEpsilon = 0;

melhorF1 = 0;

F1 = 0;

passo = (max(pval) - min(pval)) / 1000;

for epsilon = min(pval):passo:max(pval)

prev = (pval < epsilon);

vp = sum((prev==1) & (yval==1));

fp = sum((prev==1) & (yval==0));

fn = sum((prev==0) & (yval==1));

prec = vp/(vp+fp);

rec = vp/(vp+fn);

F1 = (2\*prec\*rec) / (prec+rec);

if F1 > melhorF1

melhorF1 = F1;

melhorEpsilon = epsilon;

end

end

end

Obs.: no arquivo .mlx, eu modifiquei a chamada da função para: selecionarLimiar(...); sem o Resposta no nome.

1. Copiar os valores de acuracidade, e das métricas prec, rec e F1

acuracidade = 0.1175

prec = 0.1153

rec = 1

F1 = 0.2067