

## Índice

**1. Generación de gráficas desde un CGI** 1

**2. Tarea** 2

## 1. Generación de gráficas desde un CGI

Instalación de Octave.

```
rpm -Uvh http://mirrors.kernel.org/fedora-epel/6/i386/epel-release-6-8.noarch.rpm
yum --enablerepo=epel install octave octave-forge
```

Definición de un script que llame a Octave para generar la gráfica de una variable aleatoria Gaussiana en dos dimensiones pasando el valor de la media

$$\mu = \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}$$

y los valores de matriz de covarianza:

$$C = \begin{bmatrix} C_{xx} & C_{xy} \\ C_{yx} & C_{yy} \end{bmatrix}$$

```
#!/usr/bin/octave -qf

arg_list = argv ();
mux=str2num( arg_list {1});
muy=str2num( arg_list {2});
cxx=str2num( arg_list {3});
cxy=str2num( arg_list {4});
cyy=str2num( arg_list {5});
cyx=cxy;

suffix=num2str(10000*rand());

mu=[mux;muy];
C=[cxx cxy; cyx cyy];

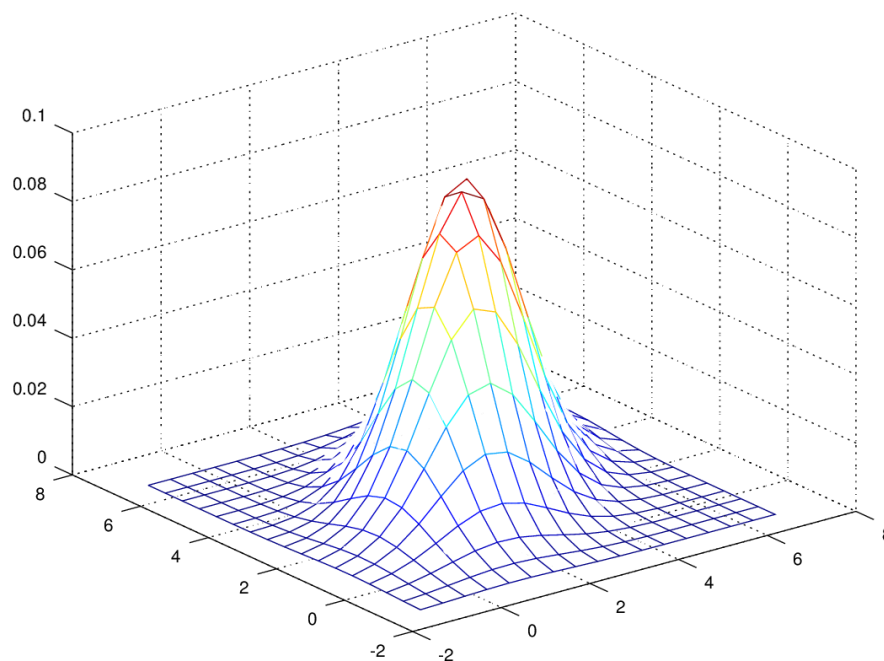
[Y,X]= ndgrid(mux-3*sqrt(cxx):0.5:mux+3*sqrt(cxx),muy-3*sqrt(cyy):0.5:muy+3*sqrt(cyy));
G = zeros(size(X,1), size(X,2));

for i=1:size(X,1)
    for j=1:size(X,2)
        D = [X(i,j);Y(i,j)];
        G(i,j) = exp( - ( (D-mu)' * (C \ (D-mu)))/2 )/sqrt((2*pi)^2*det(C));
    end
end

size(G)

figure ,mesh(X,Y,G); print ([ '/tmp/gaussian-' suffix '.png' ]); close ;
```

Ejemplo de figura generada:



## 2. Tarea

1. Adaptar el script anterior para que pueda ser la respuesta de una petición HTTP. Para ello es necesario ejecutar llamadas al sistema desde el script de Octave. A continuación se muestran algunos ejemplos:

Instrucción obtener el tamaño de un fichero desde el script de Octave:

```
[a,lon] = system("ls -al fichero.png | awk '{print $5}'",1);
```

Instrucción para escribir en la salida estándar desde el script de Octave:

```
system("printf \"HTTP/1.1 200 OK\r\n\"");
```

2. Crear una página cliente en la que se obtengan los valores a pasar al CGI.
3. Añadir el *mapping* para que cuando llegue la petición se delegue en el CGI la generación de la respuesta.