Claire Durand

ECE 3522: Stochastic Processes in Signals and Systems

Department of Electrical and Computer Engineering, Temple University, Philadelphia, PA 19122

# Problem Statement

 This assignment delved into the visualization of the Probability Distribution Function, as it is affected by covariance. The first task involved familiarizing oneself with MATLAB’s 3D visualization tools by generating 2 large independent uniformly distributed random vectors over the range [0,1] then plotting the corresponding estimate of their PDF in 3D using a function like surf. The second task involved stepping it up a notch and generating a set of Gaussian vectors according to their predefined covariance matrices

# Approach and Results

For the first task, the *rand()* function was used to defined two independent vectors, *x* and *y*. Their pdf was estimated “manually” by incrementing the elements within an output vector every time a number within the bin defined for that vector was encountered. The resulting matrix was then plotted using the *surf()* function. Figure 1 shows the resulting 3D plot. It looks noisy, but as the number of samples generated increases, the peaks should even out until they reach their expected uniform distribution.



Figure - Estimated PDF of the independent uniformly distributed random variables

Generating the Gaussian random numbers that followed a given covariance was a little bit more complicated. The *mvnrnd()* function had to be invoked to generate the appropriate random numbers. The *hist()* function was then used to estimate the individual distribution function (keeping in mind that a histogram is only a scaled version of the pdf) and *meshgrid()* was used to generate the grid over which the two pdf were to be multiplied. For the first covariance matrix, [1 0; 0 1], we expect the pdf, shown in Figure 2, to be centered and even, as the covariance matrix has 1 along the main diagonal, and 0 everywhere else.



Figure - Estimated PDF for a covariance matrix [1 0; 0 1]

The pdf of the second covariance matrix, [5 0; 0 5], shown in Figure 3, should and does yield something very similar since once again, we have even energies along the main diagonal and zeros everywhere else.



Figure - Estimated PDF for a covariance matrix [5 0; 0 5]

The pdf of the third covariance matrix, [5 0; 0 2], shown in Figure 4, should be slightly more skewed. The matrix is still symmetrical, with zeros everywhere but the main diagonal, but the elements along the main diagonal are not the same values, indicating a difference in energies. The peak of the Gaussian is slightly shifted, indicating that one of the random variables has more energy.



Figure - Estimated PDF for a covariance matrix [5 0; 0 2]

The pdf of the fourth covariance matrix, which now includes non-zero values outside of the main diagonal is shown in Figure 5. These values give the pdf a slightly more elliptical shape



Figure - Estimated PDF for a covariance matrix [1 0.5; 0.5 1]

The last covariance matrix, [5 0.5; 0.5 2], has both skew and non-zero values outside the main diagonal. Its pdf is represented in Figure 6. The pdf conserves the skew seen in Figure 4 as well as the elliptical shape seen in Figure 5.



Figure - Estimated PDF for a covariance matrix [5 0.5; 0.5 2]

# MATLAB Code

## Part 1 - Generating and plotting random vectors

maxd = 5000;
bin = 100;

z = zeros(bin,bin);

for i=1:maxd
 x(i) = rand; % Generate maxd # of random numbers between 0 and 1
 y(i) = rand;

 for n = 1:bin
 for m = 1:bin
 if((x(i)>=(1/bin)\*(m-1)) && (x(i)< (1/bin)\*m) ...
 && (y(i)>=(1/bin)\*(n-1)) && (y(i)< (1/bin)\*n))
 z(m,n) = z(m,n) + 1/maxd; % Increment and normalize
 end
 end
 end
end

figure(1)
surf(z);

## Part 2 - Generate a set of Gaussian random vectors with 2x2 cov matrix

maxd = 10000;
bin = 200;

% Define the mean
mean = [6 6];

% Define the covariance matrices
cov1 = [1 0 ; 0 1];
cov2 = [5 0 ; 0 5];
cov3 = [5 0 ; 0 2];
cov4 = [1 0.5 ; 0.5 1];
cov5 = [5 0.5 ; 0.5 2];

%z1 = zeros(bin,bin);

% Covariance 1
r1 = mvnrnd(mean, cov1, maxd); % Generate random numbers
% Estimate individual pdfs
[N1,edges1] = hist(r1(:,1),bin, 'Normalization', 'probability');
[N2,edges2] = hist(r1(:,2),bin, 'Normalization', 'probability');
[x,y] = meshgrid(N1,N2);
Z = x.\*y;
[X, Y] = meshgrid(1:1:length(Z) , 1:1:length(Z));

figure(2)
surf(X, Y, Z)
title('Estimated PDF for cov = [1 0 ; 0 1]');

% Covariance 2
r2 = mvnrnd(mean, cov2, maxd); % Generate random numbers
% Estimate individual pdfs
[N1\_2,edges1\_2] = hist(r2(:,1),bin, 'Normalization', 'probability');
[N2\_2,edges2\_2] = hist(r2(:,2),bin, 'Normalization', 'probability');
[x2,y2] = meshgrid(N1\_2,N2\_2);
Z2 = x2.\*y2; % Combine the two
[X2, Y2] = meshgrid(1:1:length(Z2) , 1:1:length(Z2)); % Create the grid

figure(3)
surf(X2, Y2, Z2)
title('Estimated PDF for cov = [5 0 ; 0 5]');

% Covariance 3
r3 = mvnrnd(mean, cov3, maxd); % Generate random numbers
% Estimate individual pdfs
[N1\_3,edges1\_3] = hist(r3(:,1),bin, 'Normalization', 'probability');
[N2\_3,edges2\_3] = hist(r3(:,2),bin, 'Normalization', 'probability');
[x3,y3] = meshgrid(N1\_3,N2\_3);
Z3 = x3.\*y3; % Combine the two
[X3, Y3] = meshgrid(1:1:length(Z3) , 1:1:length(Z3)); % Create the grid

figure(4)
surf(X3, Y3, Z3)
title('Estimated PDF for cov = [5 0 ; 0 2]');

% Covariance 4
r4 = mvnrnd(mean, cov4, maxd); % Generate random numbers
% Estimate individual pdfs
[N1\_4,edges1\_4] = hist(r4(:,1),bin, 'Normalization', 'probability');
[N2\_4,edges2\_4] = hist(r4(:,2),bin, 'Normalization', 'probability');
[x4,y4] = meshgrid(N1\_4,N2\_4);
Z4 = x4.\*y4; % Combine the two
[X4, Y4] = meshgrid(1:1:length(Z4) , 1:1:length(Z4)); % Create the grid

figure(5)
surf(X4, Y4, Z4)
title('Estimated PDF for cov = [1 0.5 ; 0.5 1]');

% Covariance 5
r5 = mvnrnd(mean, cov5, maxd); % Generate random numbers
% Estimate individual pdfs
[N1\_5,edges1\_5] = hist(r5(:,1),bin, 'Normalization', 'probability');
[N2\_5,edges2\_5] = hist(r5(:,2),bin, 'Normalization', 'probability');
[x5,y5] = meshgrid(N1\_5,N2\_5);
Z5 = x5.\*y5; % Combine the two
[X5, Y5] = meshgrid(1:1:length(Z5) , 1:1:length(Z5)); % Create the grid

figure(6)
surf(X5, Y5, Z5)
title('Estimated PDF for cov = [5 0.5 ; 0.5 2]');

[*Published with MATLAB® R2014a*](http://www.mathworks.com/products/matlab)

# Conclusions

Covariance matrices are intrinsically linked with their corresponding probability distribution function in that the covariance matrix can predict the shape of the pdf, and to some extent, vice versa. An even and symmetric covariance matrix yields a centered uniform Gaussian pdf, adding different energies along the main diagonal is reflected in a shift of the Gaussian bell, and adding non-zero values outside the main diagonal stretches the shape of the bell into an ellipse.