S7001E Lab 2

Calle Rautio
calrau-1@student.ltu.se

Jim Agnevik
jimagn-8@student.ltu.se

Contents

1 Introduction

This lab focuses on generating pseudo-random numbers in MATLAB, simulating random variables with defined probability distributions. It also investigates signal separation techniques for speech signal processing. Mixed signals were separated through correlation minimization. The lab applies these methods in a real-world scenario, where adaptive techniques are used to separate mixed newspaper audio signals.

2 Exercise 1

a) Image from 3.22 (3.16 in 3rd ed.) in the textbook. Described from the lab:

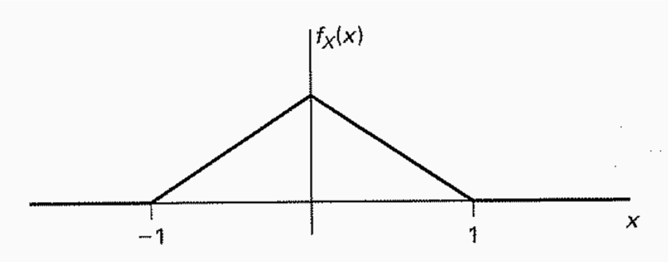
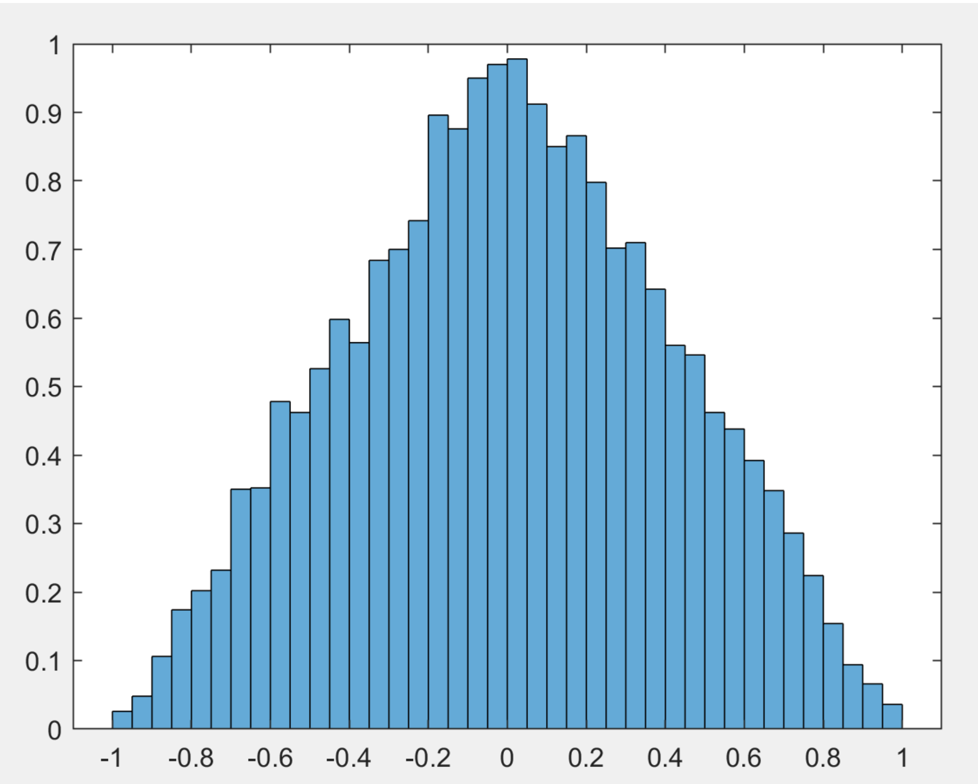
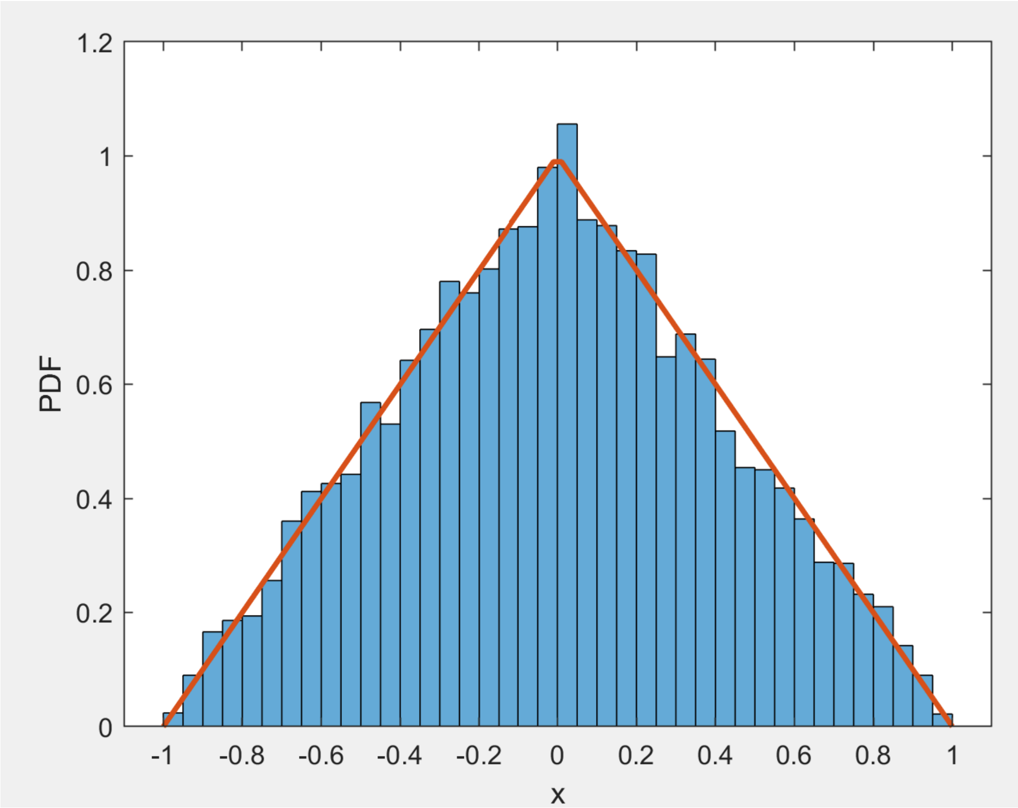


Figure P3.16

Generated image for $N = 10000$ points:



b) Generated image for $N = 10000$ points over layed with the theoretical:



Conclusion:

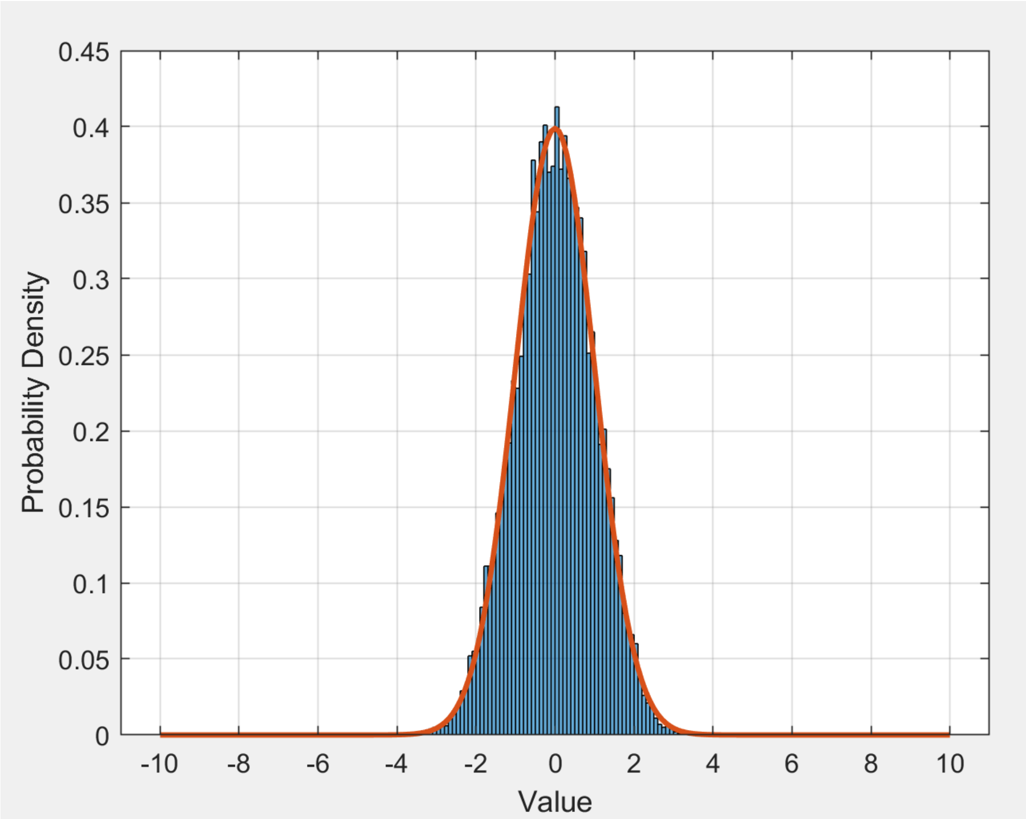
The two distributions closely align, as both follow a triangular distribution. As the value of N approaches infinity, the accuracy would increase. However, for the purposes of this lab, this demonstrates that they are effectively equivalent.

3 Exercise 2

The problem the lab wants us to solve:

3.17. It is desired to generate zero-mean Gaussian numbers. All that is available is a random number generator that generates numbers uniformly distributed on $(0, 1)$. It has been suggested Gaussian numbers might be generated by adding 12 uniformly distributed numbers and subtracting 6 from the sum. Write a program in which you use the procedure to generate 10,000 numbers and plot a histogram of your result. A histogram is a bar graph that has bins along the x-axis and number of points in the bin along the y-axis. Choose 200 bins of width 0.1 to span the range from -10 to 10. In what region of the histogram does the data look most Gaussian? Where does it look least Gaussian? Do you have any idea why this approach works?

The generated image of the Gaussian distribution displayed as a histogram, accompanied by the theoretical graph to more easily compare between the two:



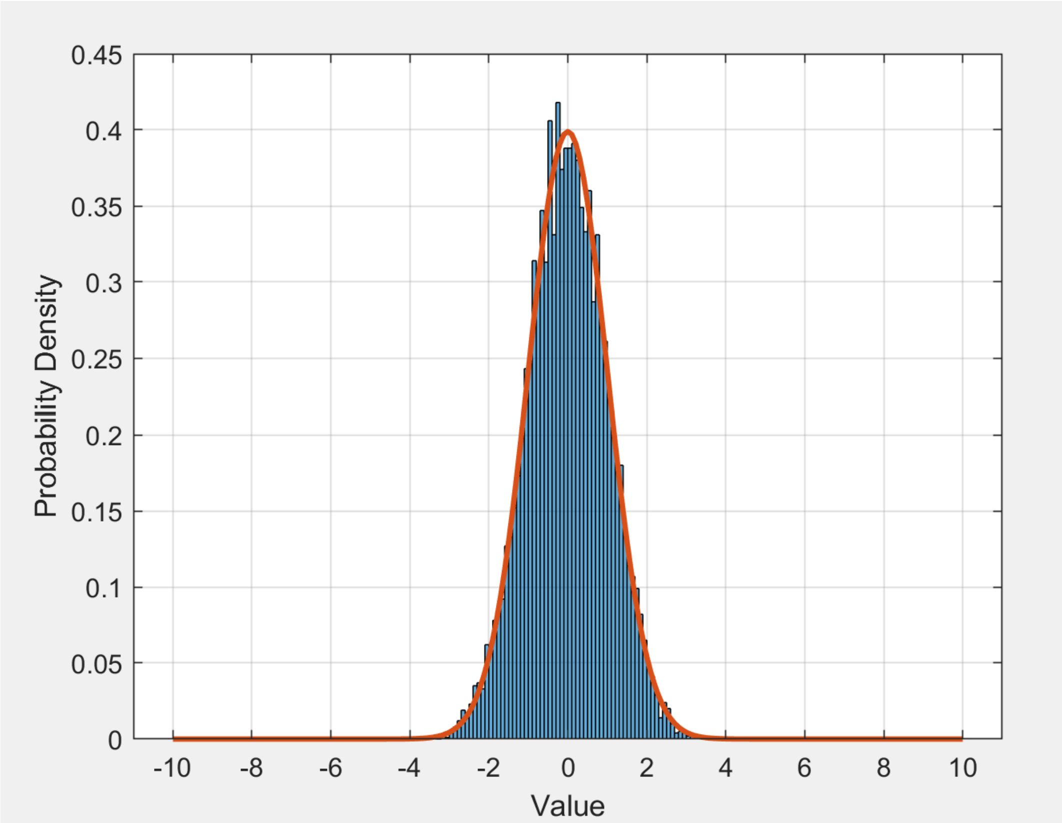
Conclusion:

The histogram closely resembles a Gaussian distribution between −4 and −1, and 1 and 4. However, between −1 and 1, it deviates most from a Gaussian distribution.

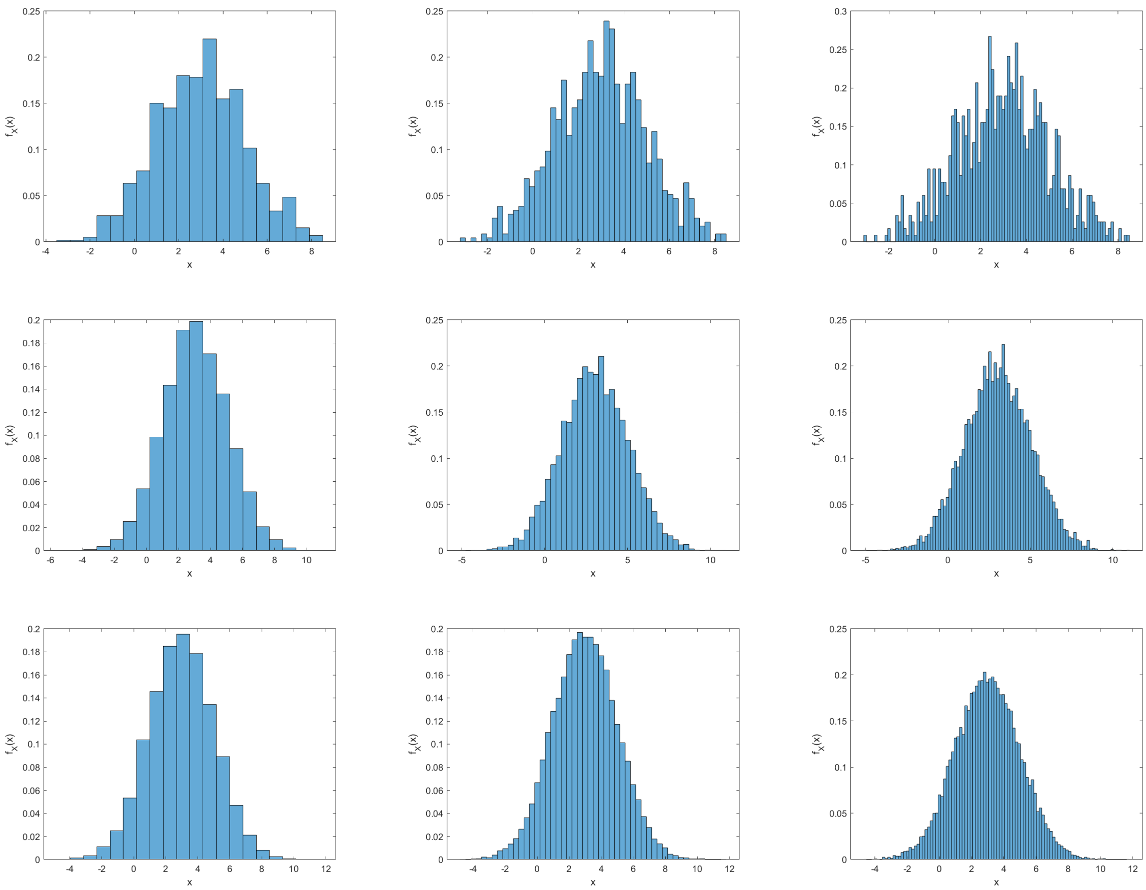
The approximation of a Gaussian distribution in this process is supported by the Central Limit Theorem. This theorem states that adding a sufficient number of independent, uniformly distributed random variables results in a sum that approximates a normal distribution. According to the theory, deviations from a perfect Gaussian distribution are most likely to occur in the tails, where finite sums of uniform random numbers might still show slight discrepancies compared to an ideal normal distribution. Although this is not apparent in our histogram, increasing the value of N would make it align more closely with the theory.

4 Exercise 3

a) Generated image for an estimate of the pdf $f_X(x)$:


0.45
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
Probability Density
-10
-8
-6
-4
-2
0
2
4
6
8
10
Value

b)


f_X(x)
x

Conclusion:

The fewer bins there are, the harder it becomes to see the Gaussian distribution. With more bins, the distribution becomes clearer. However, with a smaller N value, the observation from Exercise 2 becomes more pronounced. When having a large N and a high number of bins, the Gaussian distribution becomes increasingly clear. While a high N with a low number of bins still shows the Gaussian distribution fairly well, it's not as distinct as when both N and the number of bins are high.

5 Exercise 4

a) The expression for the joint pdf for $f_{(x_1),(x_2)}(x_1, x_2)$ is the same as $f_{x_1}(x_1) * f_{x_2}(x_2)$ because x_1 and x_2 are both independent. Meaning both together can be written as

$$f_{(x_1),(x_2)}(x_1, x_2) = \left(\frac{1}{\sqrt{2\pi * 2}} \exp\left(-\frac{(x_1-3)^2}{2*2}\right)\right) * \left(\frac{1}{\sqrt{2\pi * 1}} \exp\left(-\frac{(x_2-1)^2}{2*1}\right)\right)$$
$$(=) \frac{1}{2\pi * \sqrt{2}} \exp\left(-\frac{(x_1-3)^2}{2}\right) * \exp\left(-\frac{(x_2-1)^2}{2}\right)$$

Giving the answer for the joint pdf.

For the rest it should follow that the the mean of x_1 is μ_{x_1} = 3. The variance of x_1 is $\sigma^2_{x_1}$ = 2. The mean of x_2 is μ_{x_2} = 1 and the variance of x_2 is $\sigma^2_{x_2}$ = 1.

Since x_1 and x_2 are independent, their covariance is 0: Cov(x_1, x_2) = 0.

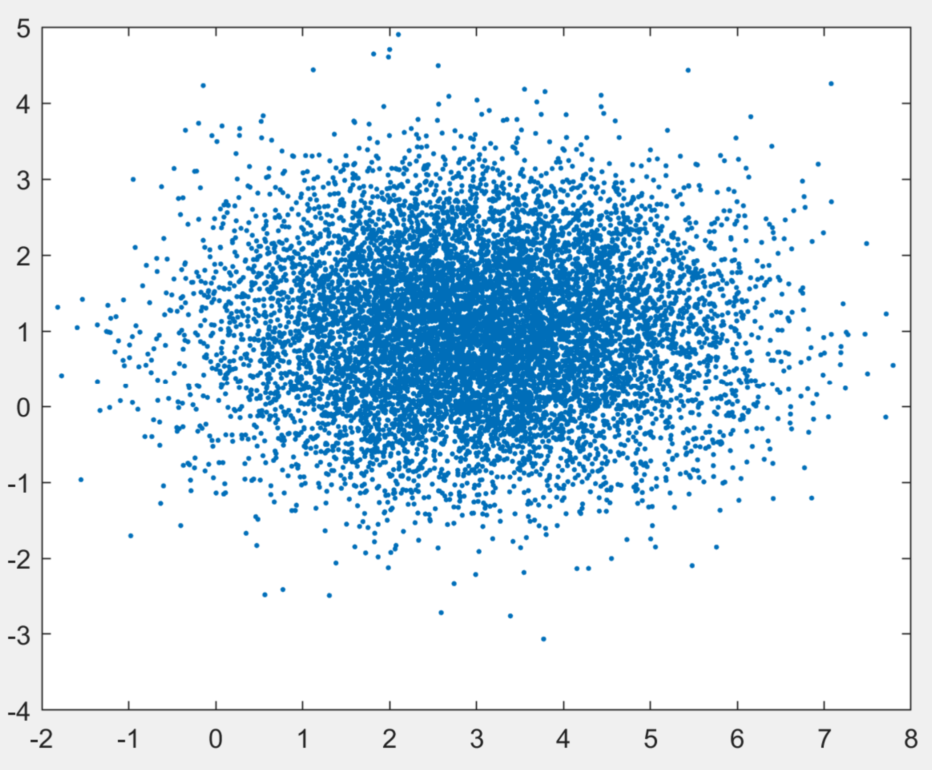
The correlation coefficient is also 0, because the correlation between two independent variables is 0. All the values were given by reading from $N(3, 2)$ and $N(1, 1)$. Because they display the mean and variance for X_1 and X_2.

b) Generated values from MATLAB:

```
Mean of X1
    2.9977

Mean of X2
    0.9941

Cov of X1 and X2
    0.0104

Corr of X1 and X2
    0.0073
```

As seen from the image the theoretical values match the ones created from MATLAB. Not perfectly but close.

c) Generated image from MATLAB of the scatter plot.



The scatter plot shows the generation of the random values generated by X_1 and X_2.

d) The expression for the joint pdf for $f_{(Y_1),(Y_2)}(Y_1, Y_2)$ =

$$f_{Y_1,Y_2}(y_1, y_2) = \frac{1}{2\pi\sqrt{|\,\Sigma_Y|}} \exp\left(-\frac{1}{2} \begin{bmatrix} y_1 - \mu_{Y_1} \\ y_2 - \mu_{Y_2} \end{bmatrix}^T \Sigma_Y^{-1} \begin{bmatrix} y_1 - \mu_{Y_1} \\ y_2 - \mu_{Y_2} \end{bmatrix} \right) \tag{1}$$

with means $\mu_{\{Y_1\}}$ and $\mu_{\{Y_2\}}$, and covariance matrix Σ_Y and where Σ_Y is the covariance matrix for Y_1 and Y_2, and $|\Sigma_Y|$ is its determinant.

To solve this we need to first solve the mean, variance, covariance and correlation.

Beginning with the mean it is the linear transformation of X_1 and X_2 means. So:

$$\mathbb{E}[Y_1] = \frac{1}{2}\mathbb{E}[X_1] + 3\mathbb{E}[X_2] \tag{2}$$

$$\mathbb{E}[Y_2] = \mathbb{E}[X_1] + \mathbb{E}[X_2] \tag{3}$$

The easiest way to get the variance is by first solving this:

$$\Sigma_Y = A\Sigma_X A^T \tag{4}$$

where Σ_X is the covariance for X_1 and X_2 which is:

$$\Sigma_X = \begin{bmatrix} \mathrm{Var}(X_1) & \mathrm{Cov}(X_1, X_2) \\ \mathrm{Cov}(X_1, X_2) & \mathrm{Var}(X_2) \end{bmatrix} \tag{5}$$

This will give the covariance matrix of Y_1 and Y_2. The diagonal entries of this matrix represent the variances $\mathrm{Var}(Y_1)$ and $\mathrm{Var}(Y_2)$, while the off-diagonal entries represent the covariance $\mathrm{Cov}(Y_1, Y_2)$.

Lastly for the correlation we can use the following formula:

$$\rho_{Y_1,Y_2} = \frac{\mathrm{Cov}(Y_1, Y_2)}{\sqrt{\mathrm{Var}(Y_1)} \cdot \sqrt{\mathrm{Var}(Y_2)}} \tag{6}$$

This is everything we need to know to solve all the variables and for the sake of this lab the answers and numbers after a bunch of combined equations will be:

The mean of Y_1 is μ_{Y_1} = 4.5. The variance of Y_1 is $\sigma^2_{Y_1}$ = 9.5.

The mean of Y_2 is μ_{Y_2} = 4 and the variance of Y_2 is $\sigma^2_{Y_2}$ = 3.

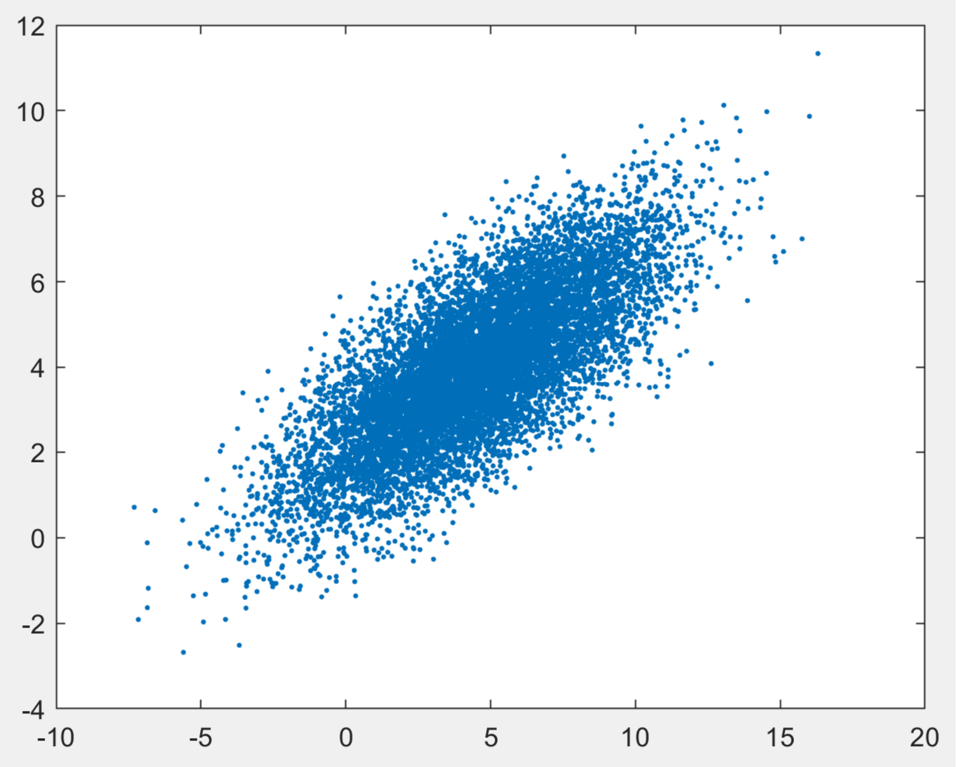
The covariance is 4

And finally the correlation is around 0.75 or to be exact 0.7492…

Generated values from MATLAB for Y_1 and Y_2. The values also closely match the theoretical.

```
Mean of Y1
    4.4811

Mean of Y2
    3.9918

Cov of Y1 and Y2
    4.0742

Corr of Y1 and Y2
    0.7513
```

Generated scatter plot from MATLAB.



The scatter plot shows the generation of the random values generated by Y_1 and Y_2.

e) With the values previously calculated we can simply use the formula for the marginal PDFs $f_{\{Y_1\}}(y_1)$ and $f_{\{Y_2\}}(y_2)$ that will be univariate Gaussian distributions:

$$f_{Y_i}(y_i) = \frac{1}{\sqrt{2\pi\sigma_{Y_i}^2}} \exp\left(-\frac{(y_i - \mu_{Y_i})^2}{2\sigma_{Y_i}^2}\right) \tag{7}$$

for $i = 1, 2$. For any given value for y_i.

f)

The conditional distribution of Y_2 given $Y_1 = y_1$ is Gaussian with the following parameters:

The mean that needs to be calculated can be done with this formula:

$$\mathbb{E}[Y_2 \mid Y_1 = y_1] = \mu_{Y_2} + \frac{\mathrm{Cov}(Y_1, Y_2)}{\mathrm{Var}(Y_1)}(y_1 - \mu_{Y_1}), \tag{8}$$

The variance that needs to be calculated can be done with this formula:

$$\mathrm{Var}(Y_2 \mid Y_1) = \mathrm{Var}(Y_2) - \frac{[\mathrm{Cov}(Y_1, Y_2)]^2}{\mathrm{Var}(Y_1)}. \tag{9}$$

Once the conditional distribution is known, the probability $P\{2 < Y_2 < 3 \mid Y_1 > 5\}$ can be computed using the conditional Gaussian distribution:

$$P\{2 < Y_2 < 3 \mid Y_1 > 5\} = \Phi\left(\frac{3 - \mu_{Y_2 \mid Y_1}}{\sigma_{Y_2 \mid Y_1}}\right) - \Phi\left(\frac{2 - \mu_{Y_2 \mid Y_1}}{\sigma_{Y_2 \mid Y_1}}\right), \tag{10}$$

where $\Phi(\cdot)$ is the CDF of the standard normal distribution. All the values should be known.

The answer after inputting all the values is $P\{2 < Y_2 < 3 \mid Y_1 > 5\} = 0.0432$

g) The value generated from MATLAB does not exactly match the theoretical and experimental values, but this could be due to numerous rounding errors across all approximations.

```
Answer for g
    0.0460
```

6 Exercise 5

6.1 a)

Given the matrix relation for the left and right channel

$$\begin{bmatrix} 1 & a \\ a & 1 \end{bmatrix} \begin{bmatrix} X \\ K \end{bmatrix} = \begin{bmatrix} L \\ R \end{bmatrix} \tag{11}$$

The simple model could be taken out, where k is a scalar.

$$\hat{X} = L + kR \tag{12}$$

$$\hat{Y} = kL + R \tag{13}$$

In order to minimize the correlation the cost function had to be minimized.

$$Q = \left(E\left[\hat{X}\hat{Y}\right]\right)^2 \tag{14}$$

By combining the equation Equation 12 Equation 13 , Equation 14 The cost function Q was determined to be Equation 15 while being made out of k, L, and R.

$$Q = (kE([L^2)] + E([LR)] + k^2E([RL)] + kE([R^2)])^2 \quad 15$$

6.2 b)

And the derivative came out as.

Derivation of Q depending on k came out as

$$\frac{dQ}{dk} = 2(2kE[LR] + E[R^2] + E([L^2)])(k^2E[LR] + k(E[R^2] + E[L^2]) + E[LR]) \quad 16$$

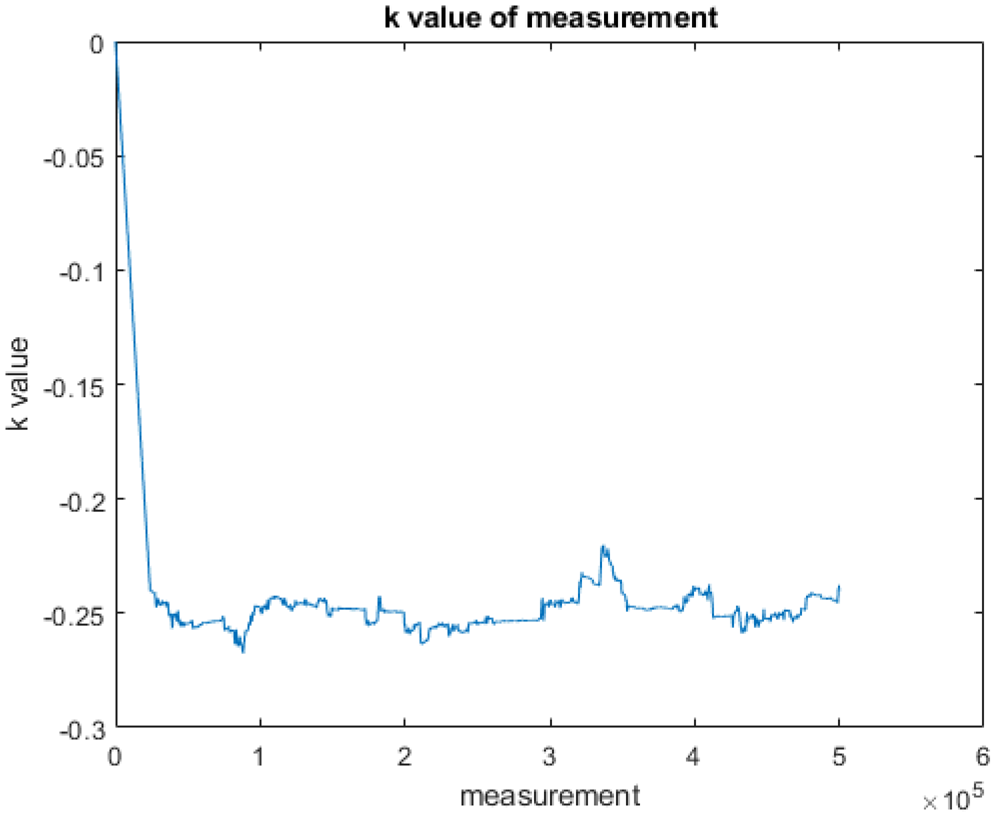
6.3 c)

Loaded in the provided audio files using [X, fs] = audioread('newspapers.wav');
as [X, fs, nbits]=wavread(newspapers.wav); was outdated

6.4 d)

Expected values was initialized from and estimation of the first 100 samples. Setting k to 0 aswell.

6.5 g)



Plot for the how the k values changes with time. Where the y axis is the k value and the x axis is the measurement for the k value over time.

6.6 h)

The Right and Left vector was saved as wav files and listened to. While listening the sound from the other channel is almost completely removed after 1 second and is a great improvement over the stereo solution. The 1 second delay coming from that the k value has not reached the correct value yet.

```
N = 10000; %Number of points
U = rand(N, 1); %Uniform random numbers

%a)
%Inverse CDF for triangular distribution
X = zeros(N, 1);
for i = 1:N
    if U(i) <= 0.5
        X(i) = -1 + sqrt(2 * U(i));
    else
        X(i) = 1 - sqrt(2 * (1 - U(i)));
    end
end

%b)
figure;
histogram(X, 'Normalization', 'pdf');
hold on;

x = linspace(-1, 1, 100);
pdf_theoretical = zeros(size(x));
for i = 1:length(x)
    if x(i) < 0
        pdf_theoretical(i) = 1 + x(i);
    else
        pdf_theoretical(i) = 1 - x(i);
    end
end

plot(x, pdf_theoretical, 'LineWidth', 2);
xlabel('x');
ylabel('PDF');
```

```
num_samples = 10000;
uniform_samples = rand(num_samples, 12);
gaussian_samples = sum(uniform_samples, 2) - 6;

bin_width = 0.1; % 20/0.1 = 200 (=) 200 bins
bin_edges = -10:bin_width:10;
histogram(gaussian_samples, 'BinEdges', bin_edges, 'Normalization', 'pdf');

hold on;

x_values = -10:0.1:10;
mu = 0;
sigma = 1;
theoretical_gaussian = (1/(sigma * sqrt(2*pi))) * exp(-((x_values - mu).^2) / (2 *
sigma^2));

plot(x_values, theoretical_gaussian, 'LineWidth', 2);

xlabel('Value');
ylabel('Probability Density');
grid on;
```

```
%a)
N = 10000;
mu = 3;
sigma = 2;
X = mu + sigma * randn(N,1);

figure;
histogram(X, 'Normalization', 'pdf');
xlabel('x');
ylabel('f_X(x)');

%b)
N_values = [1000, 10000, 20000];
bins = [20, 50, 100];
n = 0;

for N = N_values
    X = mu + sigma * randn(N, 1);

    for b = bins
        figure;
        histogram(X, b, 'Normalization', 'pdf');
        xlabel('x');
        ylabel('f_X(x)');
        %saveas(gcf,num2str(n),'png')
        n = n+1;
    end
end
```

```
n = 10000;

X1 = normrnd(3, sqrt(2), [1, n]);
X2 = normrnd(1, 1, [1, n]);

A = [1/2 3;
     1   1];

Y = A * [X1; X2];

Y1 = Y(1, :);
Y2 = Y(2, :);

%b)
disp("Mean of X1")
disp(mean(X1));
disp("Mean of X2")
disp(mean(X2));

samplecov = cov(X1,X2);
samplecorr = corrcoef(X1,X2);
disp("Cov of X1 and X2");
disp(samplecov(1,2));
disp("Corr of X1 and X2");
disp(samplecorr(1,2));

%c)
figure(1)
plot(X1,X2,'.')

%d)
disp("Mean of Y1")
disp(mean(Y1));
disp("Mean of Y2")
disp(mean(Y2));

samplecov = cov(Y1,Y2);
samplecorr = corrcoef(Y1,Y2);
disp("Cov of Y1 and Y2");
disp(samplecov(1,2));
disp("Corr of Y1 and Y2");
disp(samplecorr(1,2));

figure(2)
plot(Y1,Y2,'.')

%g)
filtered_Y2 = Y2(Y1 > 5);
prob_empirical = mean(filtered_Y2 > 2 & filtered_Y2 < 3);

disp("Answer for g")
```

```
disp(prob_empirical)
```

```
[X, fs] = audioread('newspapers.wav');

lambda = 0.99995;
Step_size = 10^-5;

L = X(:,1);
R = X(:,2);

L2=mean(X(1:100,1).^2);
R2=mean(X(1:100,2).^2);
LR=mean(X(1:100,1).*X(1:100,2));

k = zeros(length(X),1);

for i=2:length(X)

    L2=(1-lambda)*X(i-1,1)^2+lambda*L2;
    R2=(1-lambda)*X(i-1,2)^2+lambda*R2;
    LR =(1-lambda)*L(i-1)*R(i-1)+lambda*LR;

    dQdk = 2*(2*LR*k(i-1)+R2+L2)*(LR*k(i-1)^2+(R2+L2)*k(i-1)+LR);

    k(i) = k(i-1) - Step_size * sign(dQdk);
end

X_hat = L + k.*R;
Y_hat = k.*L + R;

audiowrite("sepnewspaper1.wav",X_hat/max(abs(X_hat)),fs)
audiowrite("sepnewspaper2.wav",Y_hat/max(abs(Y_hat)),fs)

plot(k)
ylabel("k value")
xlabel("measurement")
title("k value of measurement")
```