FACULTY OF INFORMATION TECHNOLOGY AND ELECTRICAL ENGINEERING
DEGREE PROGRAMME IN ELECTRONICS AND COMMUNICATIONS ENGINEERING

Statistical Signal Processing I

MATLAB Homework 3

Athmajan Vivekananthan
avivekan23@student.oulu.fi
2305573

Hathurusingha Anjalee Anuththara De Silva
hdesilva23@student.oulu.fi
2303245

1 ESTIMATION USING LINEAR MODEL

Observed two samples of a DC level in a correlated zero mean Gaussian noise with covariance matrix C are given below.

$$\begin{aligned} x[0] &= A + w[0] \\ x[1] &= A + w[1] \end{aligned} \tag{1.1}$$

where $w = [w[0]\; w[1]]^T$.

The covariance matrix C is given by:

$$C = \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$$

where $-1 \leq \rho \leq 1$ is the correlation coefficient between $w[0]$ and $w[1]$.

1.1 Derive the expression of the linear model estimator

Consider the general linear model:

$$\mathbf{x} = H\boldsymbol{\theta} + \mathbf{b} + \mathbf{w} \tag{1.2}$$

The problem given represents the linear model where;

$$\begin{aligned} H &= [1\; 1]^T \\ \boldsymbol{\theta} &= A \\ \mathbf{w} &= [w[0]\; w[1]]^T \\ \mathbf{b} &= \mathbf{0} \end{aligned}$$

Therefore the linear model equation can be reduced to :

$$\mathbf{x} = HA + \mathbf{w} \tag{1.3}$$

Since $\mathbf{w}$ is a coloured noise, a whitening matrix D can be used such that $C^{-1} = D^T D$ and the transformation $D\mathbf{w}$ results in white Gaussian noise. It can be further proved by the following :

$$\begin{aligned} \mathrm{E}[(D\mathbf{w})(\mathbf{w})^T] &= \mathrm{E}[D\mathbf{w}\mathbf{w}^T D^T] \\ &= DCD^T \\ &= DD^{-1}DD^{-1}D^{T-1}D^T \\ &= I \end{aligned} \tag{1.4}$$

This allows us to write our linear system with coloured noise with the whitening transformation:

$$\begin{aligned} \mathbf{x}' &= D\mathbf{x} \\ &= DHA + D\mathbf{w} \\ &= H'A + \mathbf{w}' \end{aligned} \tag{1.5}$$

Where $H' = DH$ and $\mathbf{w}' = D\mathbf{w}$ with $\mathbf{w}' -> N(0, I)$.

For this whitened linear model the MVU estimator is determined when the equality constraint of the CRLB theorem are satisfied. That is $\hat{A} = g(\mathbf{x}')$ will be the MVU estimate given by 1.6 for some $g(\mathbf{x}')$.

$$\frac{\partial \ln(p(\mathbf{x}'; A))}{\partial A} = I(A)(g(\mathbf{x}') - A) \tag{1.6}$$

Where $C_{\hat{A}} = I^{-1}(A)$

$$p(\mathbf{x}'; A) = \prod_{n=0}^{N-1} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[\frac{1}{2\sigma^2}(\mathbf{x}'[n] - A)^2\right] \tag{1.7}$$

$$p(\mathbf{x}'; A) = \frac{1}{(2\pi\sigma^2)^{N/2}} \exp\left[\frac{-1}{2\sigma^2}\sum_{n=0}^{N-1}(\mathbf{x}'[n] - A)^2\right] \tag{1.8}$$

$$\begin{aligned}
\frac{\partial \ln(p(\mathbf{x}'; A))}{\partial A} &= \frac{\partial}{\partial A}\left[-\ln((2\pi\sigma^2)^{N/2}) - \frac{1}{2\sigma^2}\sum_{n=0}^{N-1}(\mathbf{x}'[n] - A)^2\right] \\
&= \frac{-1}{2\sigma^2}\frac{\partial}{\partial A}\left[\mathbf{x}'^T\mathbf{x}' - 2A\mathbf{x}'^T H' + A^2 H'^T H'\right] \\
&= \frac{1}{\sigma^2}\left[H'^T\mathbf{x}' - H'^T H' A\right] \\
&= \frac{H'^T H'}{\sigma^2}\left[(H'^T H')^{-1} H'^T \mathbf{x}' - A\right]
\end{aligned} \tag{1.9}$$

Therefore the estimate and the variance of the estimate can be written as below:

$$\begin{aligned}
\hat{A} &= (H'^T H')^{-1} H'^T \mathbf{x}' \\
&= (H^T C^{-1} H)^{-1} H^T C^{-1} \mathbf{x}
\end{aligned} \tag{1.10}$$

$$\begin{aligned}
C_{\hat{A}} &= (H'^T H')^{-1} \\
&= (H^T C^{-1} H)^{-1}
\end{aligned} \tag{1.11}$$

In this derivation C is assumed to be non-singular. But according to the problem statement C becomes singular when $\rho = \pm 1$. Therefore before heading in to further analysing and simulation, the behaviour of the estimator should be investigated when ρ reaches its two possible extreme values.

1.1.1 When $\rho = \pm 1$

By using 1.10 and 1.1 the estimate for A can be opened in terms of ρ as below:

$$\begin{aligned}
\hat{A} &= \left(\begin{bmatrix}1 & 1\end{bmatrix}\frac{1}{\sigma^2}\begin{bmatrix}1 & \rho \\ \rho & 1\end{bmatrix}^{-1}\begin{bmatrix}1 \\ 1\end{bmatrix}\right)^{-1}\left(\begin{bmatrix}1 & 1\end{bmatrix}\frac{1}{\sigma^2}\begin{bmatrix}1 & \rho \\ \rho & 1\end{bmatrix}^{-1}\begin{bmatrix}x[0] \\ x[1]\end{bmatrix}\right) \\
&= \left(\begin{bmatrix}1 & 1\end{bmatrix}\frac{1}{\sigma^2(1-\rho^2)}\begin{bmatrix}1 & -\rho \\ -\rho & 1\end{bmatrix}\begin{bmatrix}1 \\ 1\end{bmatrix}\right)^{-1}\left(\begin{bmatrix}1 & 1\end{bmatrix}\frac{1}{\sigma^2(1-\rho^2)}\begin{bmatrix}1 & -\rho \\ -\rho & 1\end{bmatrix}\begin{bmatrix}x[0] \\ x[1]\end{bmatrix}\right) \\
&= \left(\frac{2}{\sigma^2(1+\rho)}\right)^{-1}\left(\frac{x[0](1-\rho) + x[1](1-\rho)}{\sigma^2(1-\rho^2)}\right) \\
&= \frac{x[0] + x[1]}{2}
\end{aligned}$$

$$
\begin{aligned}
C_{\hat{A}} &= (H^T C^{-1} H)^{-1} \\
&= \left(\begin{bmatrix} 1 & 1 \end{bmatrix} \frac{1}{\sigma^2(1-\rho^2)} \begin{bmatrix} 1 & -\rho \\ -\rho & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right)^{-1} \\
&= \frac{\sigma^2(1+\rho)}{2}
\end{aligned}
$$

Therefore since the estimate has no dependency of the ρ values we can determine the value of the estimate and the variance of the estimate at the two extemes of ρ.

Table 1.1. Estimate values at $\rho = \pm 1$

ρ values	Estimate ($\hat{A}$)	Variance C_A
+1	$\frac{x[0]+x[1]}{2}$	1
-1	$\frac{x[0]+x[1]}{2}$	0

Therefore it is clear that for the simulations we can chose the sample mean to be the estimate when the noise covariance matrix is singular.

1.2 Monte Carlo Simulation

Simulated results using Monte Carlo simulation with parameter values described below compared to theoretical values are listed in table. Matlab code used to generate the results is shown in Fig 1.1. The histograms related to each of the ρ values given in the problem statement are shown in Fig 1.2 - Fig 1.5.

It can be seen that for $\rho = -1$ the covariance matrix is singular. Therefore by investigating further it was found that the estimate was merely the sample mean since the noise is zero mean and zero variance, ergo the noise terms disappear from the linear model. Therefore the estimate is the mere exact representation of the observations.

1.2.1 Calculation of Theoretical Variances

Equation 1.9 shows that the first derivative of the log likelihood ratio with respect to the estimate. The second derivative can be derived by further partially differentiating the result from 1.9. [1]

$$
\begin{aligned}
-E\left[\frac{\partial^2 \ln(p(\mathbf{x}'; A))}{\partial A^2}\right] &= -E\frac{\partial}{\partial A}\left[-AH'^T H'\right] \\
&= H^T C^{-1} H \\
\operatorname{var}(\hat{A}) \geq \frac{1}{H^T C^{-1} H} &= \frac{\sigma^2(1+\rho)}{2}
\end{aligned} \tag{1.12}
$$

For efficient estimation the equality holds. Therefore $\operatorname{var}(\hat{A}) = \frac{1}{H^T C^{-1} H}$ Similarly the theoretical mean can be found to be equal to A since the noise is zero mean.

ρ	Theoretical Variance	Simulated Variance	Theoretical Mean	Simulated Mean
-1	0	0	2	2
0	0.5	0.5	2	7.994
0.5	0.75	0.75	2	3.5576
+1	1	1	2	1.999

Table 1.2. Simulation Results

```
% Setting up parameters of Monte Carlo simulation
% ---------------------------------------
clc
clear all
clf

numMonteCLoops = 1E5; % Number of Monte Carlo Loops
A = 2; % True value of the parameter
sigma2 = 0.5; % Parameters for covariance of noise
rhoVals = [-1, 0, 0.5,1]; % Parameters for covariance of noise
rhoInd = 2;
C = [1 rhoVals(rhoInd); rhoVals(rhoInd) 1]; % defining covariance matrix
N = 2; % Number of observations
H = [1;1]; % Observation matrix

% ---------------------------------------

% Initializing variable for stability
estimate = ones(1,numMonteCLoops)*NaN;

for MC=1:numMonteCLoops


    if rhoInd ~= 1 && rhoInd ~= 4
        % C is non-singular
        w = mvnrnd([0 0], C);
        x = H*A + w';
        Cinv = inv(C);
        invPart = H'*Cinv*H;
        estimate(MC) = invPart*H'*Cinv*x;
    elseif rhoInd == 1
        disp('rho = -1');
        % rho = -1 => C is singular but noise is reduced to ~N(0,0)
        % The estimate is the sample mean = 0.5*(x[0] + x[1])
        x = H*A + (randn(N, 1) * 0);
        estimate(MC) = mean(x);
    elseif rhoInd == 4
        disp('rho = +1');
        % rho = +1 => C is singular but noise is reduced to ~N(0,1)
        % The estimate is the sample mean = 0.5*(x[0] + x[1])
        x = H*A + (randn(N, 1) * 1);
        estimate(MC) = mean(x);

    end

end

figure(1)
clf
% Drawing the Probability Density Function of the simulated estimates
H = histogram(estimate, 'Normalization', 'pdf');
hold on
```

Figure 1.1. Matlab Code to generate simulation

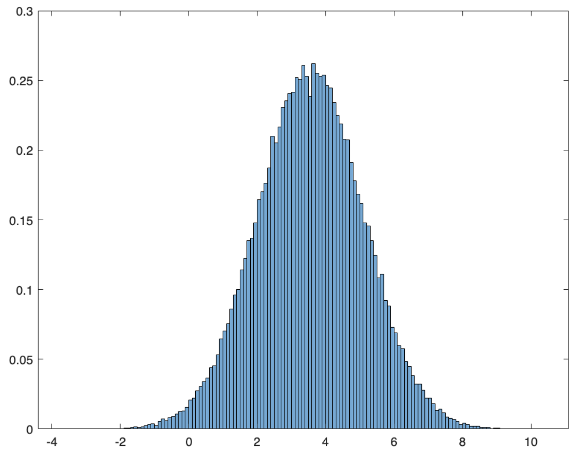


Figure 1.2. Histogram of estimated values for ρ = 0.5

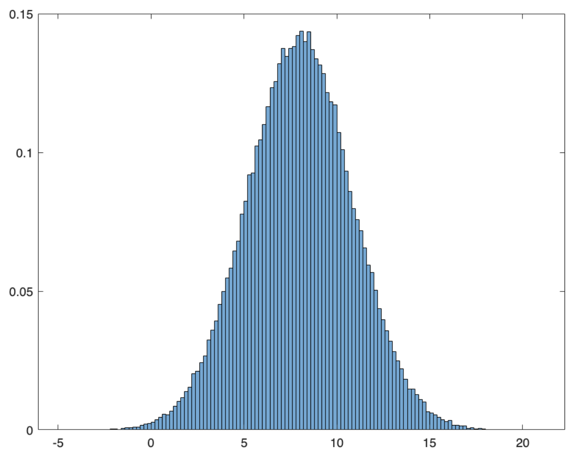


Figure 1.3. Histogram of estimated values for ρ = 0

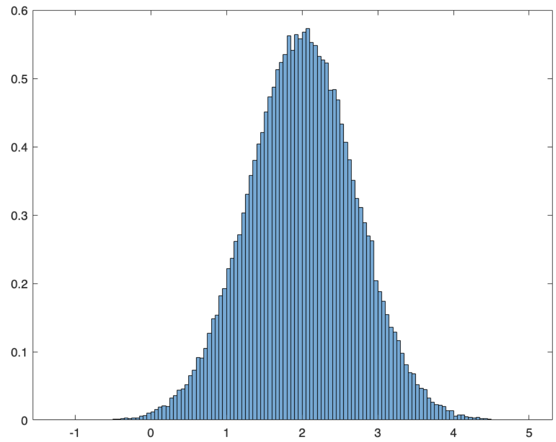


Figure 1.4. Histogram of estimated values for $\rho = 1$

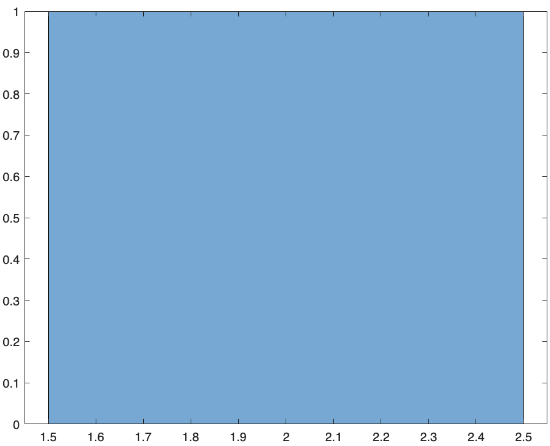


Figure 1.5. Histogram of estimated values for $\rho = -1$

2 MAXIMUM LIKELIHOOD ESTIMATOR

For N IID observations of $U[0,\theta]$ PDF (uniform distribution),

2.1 Find the maximum likelihood estimator of θ

$$p(x[n];\theta) = \begin{cases} \frac{1}{\theta} & \text{for } 0 < x[n] \leq \theta, \\ 0 & \text{otherwise.} \end{cases} \tag{2.1}$$

Since θ is independently and identically distributed the PDF of the vector $\mathbf{x}$ can be written as multiplications of individual samples of observations. [1]

$$p(\mathbf{x};\theta) = (\frac{1}{\theta})^N \tag{2.2}$$

$$\begin{aligned} \ln[p(\mathbf{x};\theta)] &= N\ln(\frac{1}{\theta}) \\ &= -N\ln(\theta) \end{aligned} \tag{2.3}$$

$$\begin{aligned} \frac{\partial \ln(p(\mathbf{x};\theta))}{\partial A} &= -N\frac{\partial \ln(\theta)}{\partial \theta} \\ &= \frac{-N}{\theta} \end{aligned} \tag{2.4}$$

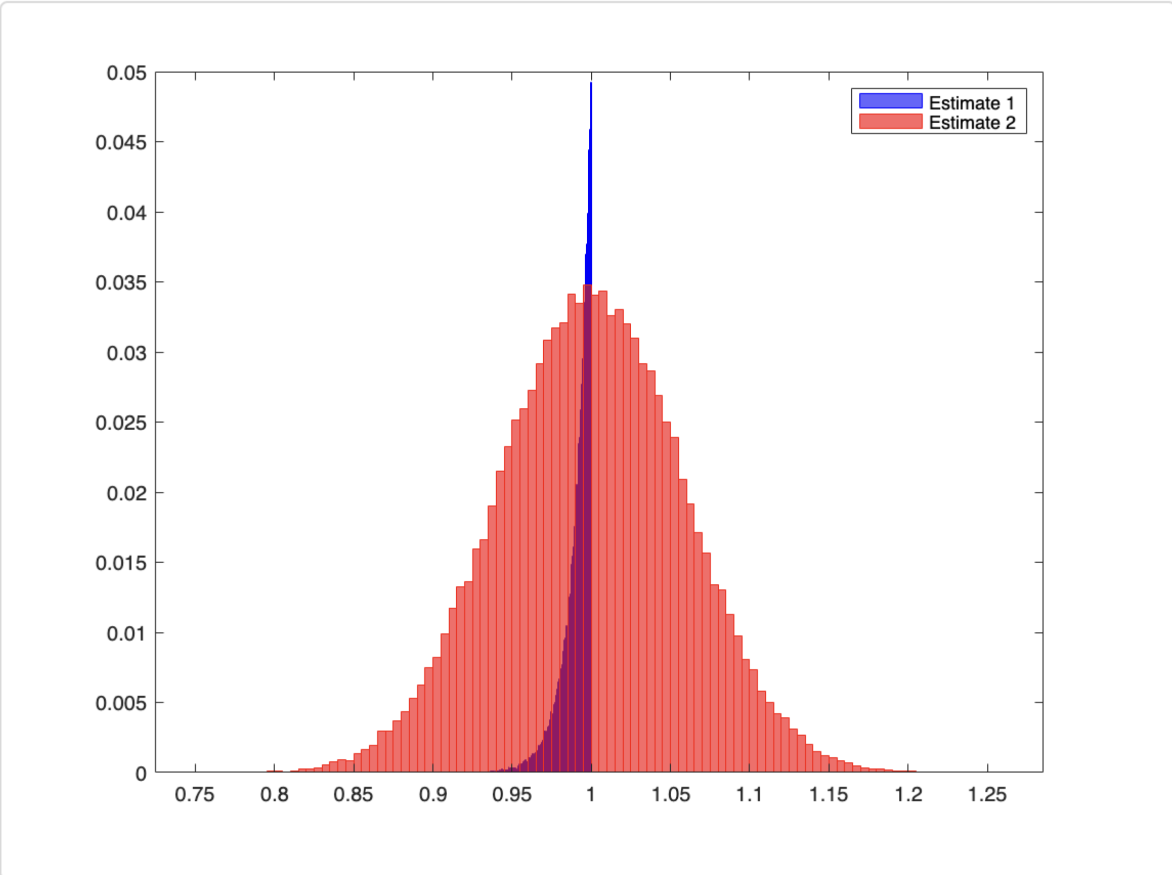
For MLE the log likelohood value should be maximized. Clearly in this case when θ increase the function is maximized. But since θ is bound between $(0,\theta]$ this maximum is achieved when $\theta = max(x[n])$. Therefore,

$$\hat{\theta} = max(x[n]) \tag{2.5}$$

2.2 Implement the maximum likelihood estimator for θ

After generating the histograms it was clear that even though the histogram of estimate 1 is skewed towards the left end, both of their expected values were very close to the true value.[2]

Therefore **it is safe to say estimate 1 is biased where as estimate 2 is unbiased**.




```
mean_est1 = 0.9901
mean_est2 = 1.0002
```

Figure 2.1. Histogram of estimates and expected values

2.3 Simulated Vs Theoretical PDFs

To calculate the theoretical PDF of estimate 1, the notation of **top order statistic** based on n observations $= x_{(n)}$ is used. It can be seen from 2.2 that the likelihood value is zero in the interval $(0, x_{(n)})$ and positive in in the interval $[x_{(n)}, \infty)$ [3]. Therefore this can be rewritten in the form :

$$\begin{aligned}\hat{\theta}(x) &= x_{(n)} \\ \hat{\theta}(X) &= X_{(n)}\end{aligned} \tag{2.6}$$

For $0 < x \leq \theta$, the distribution for $X_{(n)} = \max_{1 \leq i \leq n} X_i$ can be derived with the below arguement:

$$\begin{aligned}F_{X_n}(x) &= P\left(\max_{1 \leq i \leq n} X_i \leq x\right) \\ &= P\left(X_1 \leq x, X_2 \leq x, X_3 \leq x, \ldots, X_n \leq x\right) \\ &= P\left(X_1 \leq x\right) P\left(X_2 \leq x\right) P\left(X_3 \leq x\right) \ldots P\left(X_n \leq x\right)\end{aligned} \tag{2.7}$$

Since x are independently and identically distributed above format is possible.

$$P\left(X_i \leq x\right) = \begin{cases} 0 & \text{for } x \leq 0, \\ \frac{x}{\theta} & \text{for } 0 < x \leq \theta, \\ 1 & \text{for } \theta < x. \end{cases} \tag{2.8}$$

Therefore based on 2.7 and 2.7 the CDF of the estimator can be derived as below:

$$F_{X_n}(x) = \begin{cases} 0 & \text{for } x \leq 0, \\ \left(\frac{x}{\theta}\right)^n & \text{for } 0 < x \leq \theta, \\ 1 & \text{for } \theta < x. \end{cases} \tag{2.9}$$

By differentiating the CDF with respect to x derives the PDF of the estimate as below:

$$f_{X_n}(x) = \begin{cases} \frac{nx^{n-1}}{\theta^n} & \text{for } 0 < x \leq \theta, \\ 0 & \text{elsewhere .} \end{cases} \tag{2.10}$$

This PDF is used as the theoretical PDF in Matlab to plot against the simulated values.

Also this can be further used to proof that the **estimate 1 is unbiased**. By taking the expectation of the estimate, we can prove the bias of the estimate 1.

$$\begin{aligned}
\mathrm{E}\left[X_n\right] &= \int_0^\theta x\frac{nx^{n-1}}{\theta^n}\,dx \\
&= \frac{n}{\theta^n}\int_0^\theta x^n\,dx \\
&= \frac{n}{(n+1)\theta^n}x^{n+1}\Big|_0^\theta \\
&= \frac{n}{n+1}\theta
\end{aligned} \tag{2.11}$$

$\mathrm{E}\left[X_n\right]$ is not equal to the true value of the estimate but very very close. Therefore it proves the observation we did in the simulations where the mean was not exactly equal to the true value of the estimate.

To get the theoretical PDF of estimate 2, a Gaussian approximation is used with the mean and variance derived from the simulations. Below figure contains the plots of all PDFs in one figure. [2]

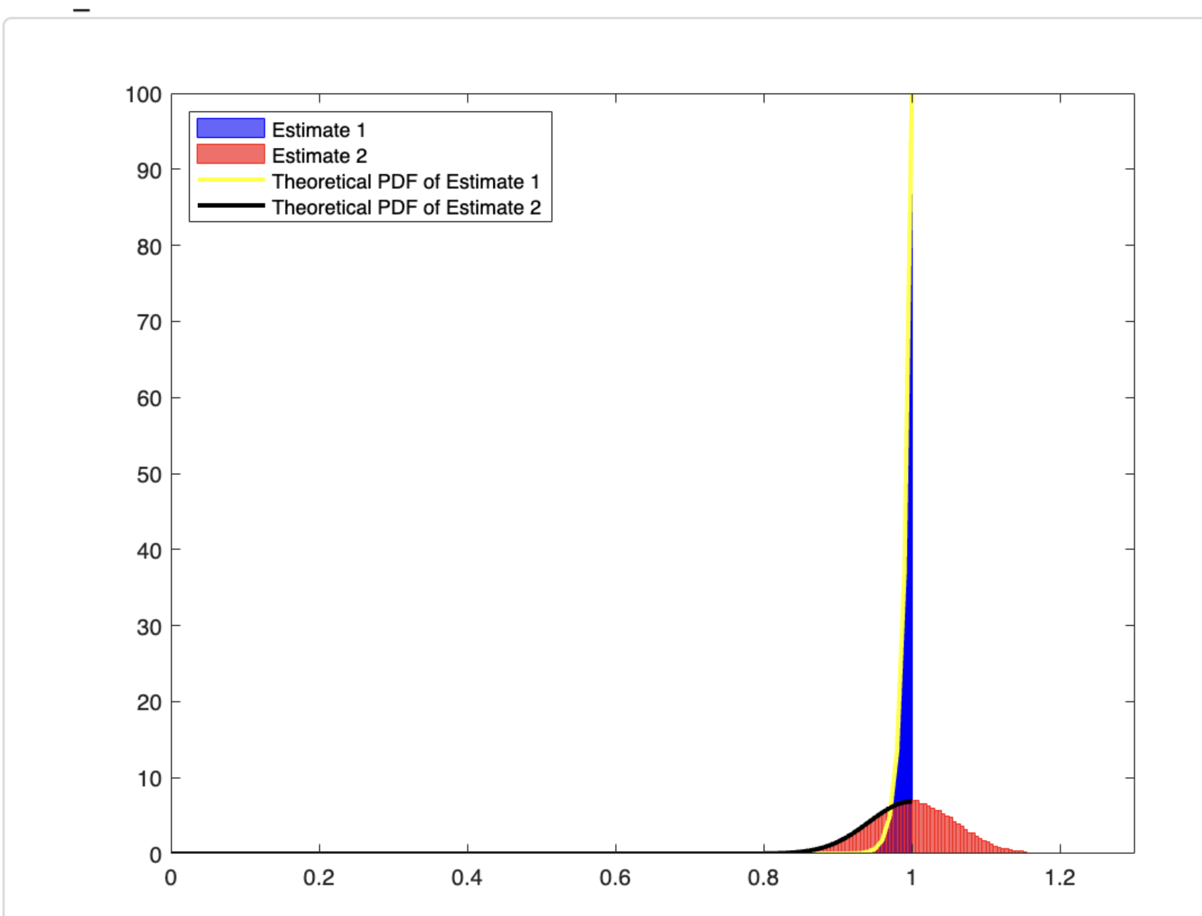


Figure 2.2. Simulated Vs Theoretical PDFs of estimate 1 and estimate 2

2.3.1 Matlab Code for task 2

```matlab
% Setting up parameters of Monte Carlo simulation
% --------------------------------------
clc
clear all
clf

numMonteCLoops = 1E5; % Number of Monte Carlo Loops
theta = 1; % True value of the parameter
N = 100; % Number of observations
% --------------------------------------


% Initializing variable for stability
estimate1 = ones(1,numMonteCLoops)*NaN;
estimate2 = ones(1,numMonteCLoops)*NaN;
for MC=1:numMonteCLoops
    x = rand(theta, N);
    estimate1(MC) = max(x);

    A_est_mean = mean(x)*2;
    estimate2(MC) = A_est_mean;
end


hist1 = histogram(estimate1, 'Normalization', 'pdf', 'EdgeColor', 'b', 'FaceColor', 'b', 'DisplayName', 'Estimate 1');
hold on;
histogram(estimate2, 'Normalization', 'pdf', 'EdgeColor', 'r', 'FaceColor', 'r', 'DisplayName', 'Estimate 2');

legend('show');
hold on;

% Plot theoretical value of estimate 1
% Proof of theoretical PDF is provided in the report.
x = 0:0.01:1;
theoPDF_est1 = N * x.^(N - 1) / theta^N;
theoPDF_est1_normalized =max(estimate1) * theoPDF_est1 / max(theoPDF_est1);
plot(x, theoPDF_est1, 'LineWidth', 2, 'DisplayName', 'Theoretical PDF of Estimate 1','Color', 'yellow');
hold on;

sigma_est2 = std(estimate2);
mu = mean(estimate2);
theoPDF_est2 = 1 / (sigma_est2 * sqrt(2*pi)) * exp(-0.5 * ((x - mu) / sigma_est2).^2);
plot(x, theoPDF_est2, 'LineWidth', 2, 'DisplayName', 'Theoretical PDF of Estimate 2','Color', 'black');

legend('Location', 'northwest');
```

Figure 2.3. Matlab Code Screenshot for task 2

3 REFERENCES

[1] Kay S.M. (1993) Fundamentals of Statistical Signal Processing, Volume 1: Estimation Theory. Pearson Education.

[2] Mathworks (2023) Histogram plot. Technical report, Mathworks, https://www.mathworks.com/help/matlab/ref/matlab.graphics.chart.primitive.histogram.html (2023).

[3] Muse R. (2019) Introduction to Statistical Methods. University of Arizona.