Exercise 1: parametric estimation of a resistance from current–voltage data

Consider a resistor described by the following current–voltage instantaneous relationship:

$$V_j = R \cdot i_j + e_j, \quad j = 1, 2, \ldots, N \qquad \Leftrightarrow \qquad \begin{bmatrix} V_1 \\ \vdots \\ V_N \end{bmatrix} = \begin{bmatrix} i_1 \\ \vdots \\ i_N \end{bmatrix} \cdot R + \begin{bmatrix} e_1 \\ \vdots \\ e_N \end{bmatrix}$$

where R is the resistance, i is the current in the resistor, V is the voltage drop across the resistor terminals and e is a white noise with zero mean value and variance matrix Σ_{ee}. Two different datasets are available, each containing the following data vectors:

- $i = [i_1 \ \cdots \ i_N]^T$: $N = 20$ current measurements equispaced in the interval $[1, 10]\, A$;
- $V = [V_1 \ \cdots \ V_N]^T$: $N = 20$ noise-corrupted voltage measurements corresponding to the above current values.

Problem: For both the datasets, compute the following estimates of the resistance R:

- Sample Mean estimate $\hat{R}_{SM}$;
- Least Squares estimate $\hat{R}_{LS}$;
- Gauss-Markov estimate $\hat{R}_{GM}$;
- Bayesian estimate $\hat{R}_B$.

Evaluate the quality of the obtained estimates, by comparing on a plot the measured and the "predicted" voltages and by comparing the estimate variances of $\hat{R}_{LS}$, $\hat{R}_{GM}$ and $\hat{R}_B$.

Main steps:

(1) Load the data from the file `resistor_data_1.mat` (use `load` MATLAB command).

(2) Plot on a figure the measured data (use `plot` MATLAB command).

(3) Compute the Sample Mean estimate $\hat{R}_{SM}$ (use `mean` MATLAB command). Plot in the figure generated at step (2) the estimated resistor characteristic $\hat{V}_{SM} = \hat{R}_{SM} \cdot i$.

(4) Compute the Least Squares estimate $\hat{R}_{LS}$ considering the noise variance matrix $\Sigma_{ee} = \sigma_e^2 I_N$, where $\sigma_e^2 = 25\ Volt^2$ and I_N is the identity matrix (use `mldivide` or \ or `pinv` MATLAB command). Plot in the figure generated at steps (2)–(3) the estimated resistor characteristic $\hat{V}_{LS} = \hat{R}_{LS} \cdot i$.

(5) Compute the Gauss-Markov estimate $\hat{R}_{GM}$. Plot in the figure generated at steps (2)–(4) the estimated resistor characteristic $\hat{V}_{GM} = \hat{R}_{GM} \cdot i$.

(6) Considering R as a scalar random variable uncorrelated with the noise sequence e (i.e., $\Sigma_{Re} = E[(R - \overline{R})e^T] = \mathbf{0}_{1\times N}$), compute the Bayesian estimate $\hat{R}_B$:

$$\hat{R}_B = \overline{R} + \Sigma_{RV}\Sigma_{VV}^{-1}(V - \overline{V})$$

where:

- $\overline{R} = E[R] =$ *a priori* mean value of R, to be assumed early (try with a value around 3);
- $\Sigma_{RR} = E[(R - \overline{R})^2] = \sigma_R^2 =$ *a priori* variance of R, to be assumed early (try with the following different values of σ_R^2: 10, 1, 0.1, 0.01);
- $\overline{V} = E[V] = E[R \cdot i + e] = E[R \cdot i] + E[e] = E[R] \cdot i = \overline{R} \cdot i$;
- $\Sigma_{VV} = E\left[(V - \overline{V})(V - \overline{V})^T\right] = E\left[(R \cdot i + e - \overline{R} \cdot i)(R \cdot i + e - \overline{R} \cdot i)^T\right]$
  $= E\left[((R - \overline{R}) \cdot i + e)((R - \overline{R}) \cdot i + e)^T\right]$
  $= E\left[((R - \overline{R}) \cdot i)((R - \overline{R}) \cdot i)^T\right] + E\left[((R - \overline{R}) \cdot i)e^T\right] + E\left[e((R - \overline{R}) \cdot i)^T\right] + E\left[ee^T\right]$
  $= E\left[(R - \overline{R})^2\right] \cdot i \cdot i^T + i \cdot E\left[(R - \overline{R})e^T\right] + E\left[e(R - \overline{R})\right] \cdot i^T + E\left[ee^T\right]$
  $= \Sigma_{RR} \cdot i \cdot i^T + i \cdot \Sigma_{Re} + \Sigma_{eR} \cdot i^T + \Sigma_{ee} = \sigma_R^2 \cdot i \cdot i^T + \Sigma_{ee}$;
- $\Sigma_{RV} = E\left[(R - \overline{R})(V - \overline{V})^T\right] = E\left[(R - \overline{R})(R \cdot i + e - \overline{R} \cdot i)^T\right]$
  $= E\left[(R - \overline{R})(R \cdot i - \overline{R} \cdot i)^T + (R - \overline{R})e^T\right] = E\left[(R - \overline{R})(R \cdot i - \overline{R} \cdot i)^T\right] + E\left[(R - \overline{R})e^T\right]$
  $= E\left[(R - \overline{R})(R - \overline{R}) \cdot i^T\right] + \Sigma_{Re} = E\left[(R - \overline{R})^2\right] \cdot i^T = \sigma_R^2 \cdot i^T$.

Plot in the figure generated at steps (2)–(5) the estimated resistor characteristic $\hat{V}_B = \hat{R}_B \cdot i$.

(7) Compute and compare the variances of the estimates $\hat{R}_{LS}$, $\hat{R}_{GM}$ and $\hat{R}_B$.

(8) Repeat the steps (1)–(7) with the data stored in the file `resistor_data_2.mat`, using in steps (4)-(7) a noise variance matrix Σ_{ee} consistent with the data.

Possible solution under MATLAB (file Lab3_E1A.m)

```
%% Laboratory 3 - Estimation, filtering and system identification - Prof. M. Taragna
% *Exercise 1: Parametric estimation of a resistance from current-voltage data*
% *stored in |resistor_data_1.mat|*
%% Introduction
% The program code may be splitted in sections using the characters "%%".
% Each section can run separately with the command "Run Section"
% (in the Editor toolbar, just to the right of the "Run" button). You can do the
% same thing by highlighting the code you want to run and by using the button
% function 9 (F9). This way, you can run only the desired section of your code,
% saving your time. This script can be considered as a reference example.

clear all, close all, clc

%% Procedure
% # Load the file |resistor_data_1.mat| containing the current-voltage data
% # Plot the voltage measurements versus the current measurements
% # Estimate the resistance using the Sample Mean algorithm
% # Plot the estimated approximation versus the experimental data
% # Estimate the resistance using the Least Squares algorithm
% # Plot the estimated approximation versus the experimental data
% # Estimate the resistance using the Gauss-Markov algorithm
% # Plot the estimated approximation versus the experimental data
% # Estimate the resistance using the Bayesian method
% # Plot the estimated approximation versus the experimental data

%% Problem setup

% Step 1: load of data

load resistor_data_1.mat
% i = current measured in Ampere
% V = voltage measured in Volt

N = length(i); % N = number of data

% Step 2: plot of data

figure, plot(i,V,'ok'),
axis([min(i),max(i),0,35]), grid, title('Measured data and approximations'),
xlabel('Current{\it i} in A'), ylabel('Voltage{\it V} in V'),
legend('Measured data',4)

%% Estimation of the resistance using the Sample Mean algorithm

% Step 3: computation of the parameter estimate as mean of the R values
% derived from the Ohm's law for each couple of current-voltage data

R_SM = mean(V./i)   % Form #1: using the "mean" operator

R_SM_ = V'*(1./i)/N % Form #2: using the Sample Mean definition

% Step 4: graphical comparison of the results

V_SM = R_SM*i;
figure, plot(i,V,'ok', i,V_SM,'-b'),
axis([min(i),max(i),0,35]), grid, title('Measured data and approximations'),
xlabel('Current{\it i} in A'), ylabel('Voltage{\it V} in V'),
legend('Measured data','Sample Mean estimate',4)

%% Estimation of the resistance using the Least Squares algorithm

% Step 5: computation of the parameter estimate

Phi=i;
R_LS=Phi\V   % Form #1: using the "\" operator (more realiable)
```

```
A=pinv(Phi); % more realiable than inv(Phi'*Phi)*Phi'
R_LS_=A*V    % Form #2: using the pseudoinverse matrix
sigma_e = 5;
Sigma_ee = sigma_e^2*eye(N);
Sigma_R_LS = pinv(Phi)*Sigma_ee*pinv(Phi)'

% Step 6: graphical comparison of the results

V_LS = R_LS*i;
figure, plot(i,V,'ok', i,V_SM,'-b', i,V_LS,'-r'),
axis([min(i),max(i),0,35]), grid, title('Measured data and approximations'),
xlabel('Current{\it i} in A'), ylabel('Voltage{\it V} in V'),
legend('Measured data','Sample Mean estimate','Least Squares estimate',4)

%% Estimation of the resistance using the Gauss-Markov algorithm

% Step 7: computation of the parameter estimate
% Note that the Gauss-Markov estimate coincides with the Least Squares
% estimate when the same weights are used for all the measurements

R_GM = inv(Phi'*inv(Sigma_ee)*Phi)*Phi'*inv(Sigma_ee)*V
Sigma_R_GM = inv(Phi'*inv(Sigma_ee)*Phi)

% Step 8: graphical comparison of the results

V_GM = R_GM*i;
figure, plot(i,V,'ok', i,V_SM,'-b', i,V_LS,'-r', i,V_GM,'--k'),
axis([min(i),max(i),0,35]), grid, title('Measured data and approximations'),
xlabel('Current{\it i} in A'), ylabel('Voltage{\it V} in V'),
legend('Measured data','Sample Mean estimate','Least Squares estimate',...
       'Gauss-Markov estimate',4)

%% Estimation of the resistance using the Bayesian method

% Step 9: computation of the parameter estimate, assuming:
% - R_bar = a priori mean value of R = 3 (to be assumed early)
% - Sigma_RR = a priori variance of R = 10, 1, 0.1, 0.01 (to be assumed early)
% - V_bar = mean value of V = R_bar*i
% - Sigma_VV = Sigma_RR*i*i' + Sigma_ee
% - Sigma_RV = Sigma_RR*i'

R_bar = 3; % Try also with R_GM
Sigma_RR = [10, 1, 0.1, 0.01];
V_bar = R_bar*i;
for k=1:length(Sigma_RR),
    Sigma_VV = Sigma_RR(k)* i*i' + Sigma_ee;
    Sigma_RV = Sigma_RR(k)*i';
    R_B(k) = R_bar + Sigma_RV * inv(Sigma_VV) * (V-V_bar);
    Sigma_R_B(k) = Sigma_RV * inv(Sigma_VV) * Sigma_RV'; % estimate variance
    Sigma_R_minus_R_B(k) = Sigma_RR(k) - Sigma_R_B(k); % a posteriori variance
    V_B(:,k) = R_B(k)*i;
end
Sigma_RR, R_B, Sigma_R_B, Sigma_R_minus_R_B

% Step 10: graphical comparison of the results

figure, plot(i,V,'ok', i,V_SM,'-b', i,V_LS,'-r', i,V_GM,'--k', ...
             i,V_B(:,1),'-m', i,V_B(:,2),'--m', ...
             i,V_B(:,3),'-c', i,V_B(:,4),'--c'),
axis([min(i),max(i),0,35]), grid, title('Measured data and approximations'),
xlabel('Current{\it i} in A'), ylabel('Voltage{\it V} in V'),
legend('Measured data','Sample Mean estimate','Least Squares estimate',...
       'Gauss-Markov estimate', ...
       ['Bayesian estimate (\sigma_R^2=', num2str(Sigma_RR(1)),')'], ...
       ['Bayesian estimate (\sigma_R^2=', num2str(Sigma_RR(2)),')'], ...
       ['Bayesian estimate (\sigma_R^2=', num2str(Sigma_RR(3)),')'], ...
       ['Bayesian estimate (\sigma_R^2=', num2str(Sigma_RR(4)),')'],4)
```

Possible solution under MATLAB (file Lab3_E1B.m)

```
%% Laboratory 3 - Estimation, filtering and system identification - Prof. M. Taragna
% *Exercise 1: Parametric estimation of a resistance from current-voltage data*
% *stored in |resistor_data_2.mat|*
%% Introduction
% The program code may be splitted in sections using the characters "%%".
% Each section can run separately with the command "Run Section"
% (in the Editor toolbar, just to the right of the "Run" button). You can do the
% same thing by highlighting the code you want to run and by using the button
% function 9 (F9). This way, you can run only the desired section of your code,
% saving your time. This script can be considered as a reference example.

clear all, close all, clc

%% Procedure
% # Load the file |resistor_data_2.mat| containing the current-voltage data
% # Plot the voltage measurements versus the current measurements
% # Estimate the resistance using the Sample Mean algorithm
% # Plot the estimated approximation versus the experimental data
% # Define the noise variance matrix consistent with experimental data
% # Estimate the resistance using the Least Squares algorithm
% # Plot the estimated approximation versus the experimental data
% # Estimate the resistance using the Gauss-Markov algorithm
% # Plot the estimated approximation versus the experimental data
% # Estimate the resistance using the Bayesian method
% # Plot the estimated approximation versus the experimental data

%% Problem setup

% Step 1: load of data

load resistor_data_2.mat
% i = current measured in Ampere
% V = voltage measured in Volt

N = length(i); % N = number of data

% Step 2: plot of data

figure, plot(i,V,'ok'),
axis([min(i),max(i),0,45]), grid, title('Measured data and approximations'),
xlabel('Current{\it i} in A'), ylabel('Voltage{\it V} in V'),
legend('Measured data',4)

%% Estimation of the resistance using the Sample Mean algorithm

% Step 3: computation of the parameter estimate as mean of the R values
% derived from the Ohm's law for each couple of current-voltage data

R_SM = mean(V./i)   % Form #1: using the "mean" operator

R_SM_ = V'*(1./i)/N % Form #2: using the Sample Mean definition

% Step 4: graphical comparison of the results

V_SM = R_SM*i;
figure, plot(i,V,'ok', i,V_SM,'-b'),
axis([min(i),max(i),0,45]), grid, title('Measured data and approximations'),
xlabel('Current{\it i} in A'), ylabel('Voltage{\it V} in V'),
legend('Measured data','Sample Mean estimate',4)

%% Estimation of the resistance using the Least Squares algorithm

% Step 5: definition of the noise variance matrix consistent with data

sigma_e = 5;
Sigma_ee = sigma_e^2*eye(N); % Like with resistor_data_1.mat datafile
```

```
% From data plot, the 7-th sample is not consistent with the others,
% being highly corrupted. As a consequence, the weight of this sample
% should be reduced. This can be obtained by forcing a high value of
% the corresponding variance in the noise variance matrix, since each
% variance represents the measure uncertainty: a larger variance reduces
% the measure reliability. For example, instead of sigma_e^2 = 5^2,
% the value (5*sigma_e)^2 = 25^2 can be forced.
Sigma_ee(7,7) = (5*sigma_e)^2;

% Step 6: computation of the parameter estimate

Phi=i;
R_LS=Phi\V   % Form #1: using the "\" operator (more realiable)

A=pinv(Phi); % more realiable than inv(Phi'*Phi)*Phi'
R_LS_=A*V    % Form #2: using the pseudoinverse matrix

Sigma_R_LS = pinv(Phi)*Sigma_ee*pinv(Phi)'

% Step 7: graphical comparison of the results

V_LS = R_LS*i;
figure, plot(i,V,'ok', i,V_SM,'-b', i,V_LS,'-r'),
axis([min(i),max(i),0,45]), grid, title('Measured data and approximations'),
xlabel('Current{\it i} in A'), ylabel('Voltage{\it V} in V'),
legend('Measured data','Sample Mean estimate','Least Squares estimate',4)

%% Estimation of the resistance using the Gauss-Markov algorithm

% Step 8: computation of the parameter estimate
% Note that the Gauss-Markov estimate differs from the Least Squares
% estimate when different weights are used for the measurements

R_GM = inv(Phi'*inv(Sigma_ee)*Phi)*Phi'*inv(Sigma_ee)*V
Sigma_R_GM = inv(Phi'*inv(Sigma_ee)*Phi)

% Step 9: graphical comparison of the results

V_GM = R_GM*i;
figure, plot(i,V,'ok', i,V_SM,'-b', i,V_LS,'-r', i,V_GM,'--k'),
axis([min(i),max(i),0,45]), grid, title('Measured data and approximations'),
xlabel('Current{\it i} in A'), ylabel('Voltage{\it V} in V'),
legend('Measured data','Sample Mean estimate','Least Squares estimate',...
       'Gauss-Markov estimate',4)

%% Estimation of the resistance using the Bayesian method

% Step 10: computation of the parameter estimate, assuming:
% - R_bar = a priori mean value of R = 3 (to be assumed early)
% - Sigma_RR = a priori variance of R = 10, 1, 0.1, 0.01 (to be assumed early)
% - V_bar = mean value of V = R_bar*i
% - Sigma_VV = Sigma_RR*i*i' + Sigma_ee
% - Sigma_RV = Sigma_RR*i'

R_bar = 3; % Try also with R_GM
Sigma_RR = [10, 1, 0.1, 0.01];
V_bar = R_bar*i;
for k=1:length(Sigma_RR),
    Sigma_VV = Sigma_RR(k)* i*i' + Sigma_ee;
    Sigma_RV = Sigma_RR(k)*i';
    R_B(k) = R_bar + Sigma_RV * inv(Sigma_VV) * (V-V_bar);
    Sigma_R_B(k) = Sigma_RV * inv(Sigma_VV) * Sigma_RV'; % estimate variance
    Sigma_R_minus_R_B(k) = Sigma_RR(k) - Sigma_R_B(k); % a posteriori variance
    V_B(:,k) = R_B(k)*i;
end
Sigma_RR, R_B, Sigma_R_B, Sigma_R_minus_R_B
```

```
% Step 11: graphical comparison of the results

figure, plot(i,V,'ok', i,V_SM,'-b', i,V_LS,'-r', i,V_GM,'--k', ...
             i,V_B(:,1),'-m', i,V_B(:,2),'--m', ...
             i,V_B(:,3),'-c', i,V_B(:,4),'--c'),
axis([min(i),max(i),0,45]), grid, title('Measured data and approximations'),
xlabel('Current{\it i} in A'), ylabel('Voltage{\it V} in V'),
legend('Measured data','Sample Mean estimate','Least Squares estimate',...
       'Gauss-Markov estimate', ...
       ['Bayesian estimate (\sigma_R^2=', num2str(Sigma_RR(1)),')'], ...
       ['Bayesian estimate (\sigma_R^2=', num2str(Sigma_RR(2)),')'], ...
       ['Bayesian estimate (\sigma_R^2=', num2str(Sigma_RR(3)),')'], ...
       ['Bayesian estimate (\sigma_R^2=', num2str(Sigma_RR(4)),')'],4)
```