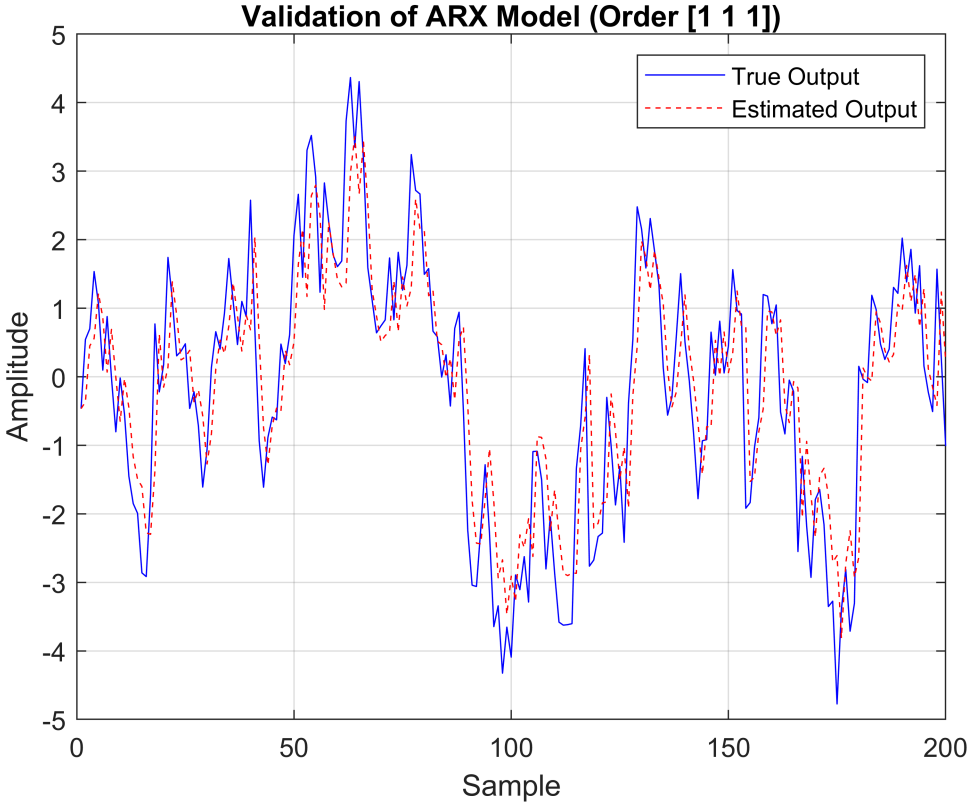
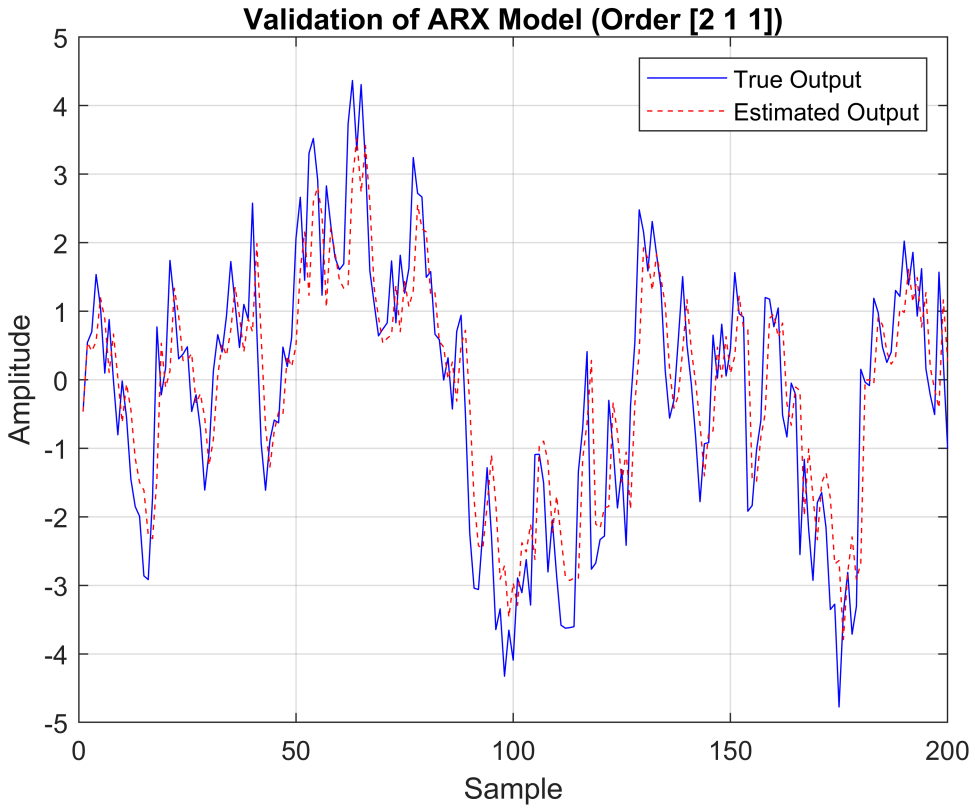
```
% Define the system (original transfer function)
numerator = [1, -0.8];
denominator = conv(conv([1, -0.1], [1, -0.5]), [1, -0.94]); % Convolve denominator factors
G = tf(numerator, denominator, 1); % Transfer function with a sampling time of 1

% Generate input signal
N = 1000; % Number of data points
u = randn(N, 1); % Input signal (random Gaussian noise)

% Simulate system output
y = lsim(G, u); % Simulate the response of the system

% Split data for system identification
% Use first 80% of the data for training, rest for validation
split_idx = round(0.8 * N);
u_train = u(1:split_idx);
y_train = y(1:split_idx);
u_val = u(split_idx+1:end);
y_val = y(split_idx+1:end);

% Create iddata object for training
data_train = iddata(y_train, u_train, 1); % Sampling time = 1

% Create iddata object for validation
data_val = iddata(y_val, u_val, 1);

% Define ARX model orders for first-order, second-order, and third-order systems
orders = [
    1, 1, 1;  % First-order system: [na nb nk]
    2, 1, 1;  % Second-order system: [na nb nk]
    3, 1, 1   % Third-order system: [na nb nk]
];

% Loop through the model orders and estimate ARX models
for i = 1:size(orders, 1)
    na = orders(i, 1); % Denominator order
    nb = orders(i, 2); % Numerator order
    nk = orders(i, 3); % Input-output delay

    % Estimate ARX model
    model_arx = arx(data_train, [na nb nk]);

    % Display the ARX model
    fprintf('Estimated ARX model for order [%d %d %d]:\n', na, nb, nk);
    disp(model_arx);

    % Validate the model using validation data
    y_est = predict(model_arx, data_val);

    % Plot comparison between true and estimated output
    figure;
    plot(y_val, 'b'); hold on; % True output (validation data)
```

```
    plot(y_est.OutputData, 'r--'); % Estimated output
    legend('True Output', 'Estimated Output');
    xlabel('Sample');
    ylabel('Amplitude');
    title(sprintf('Validation of ARX Model (Order [%d %d %d])', na, nb, nk));
    grid on;
end
```

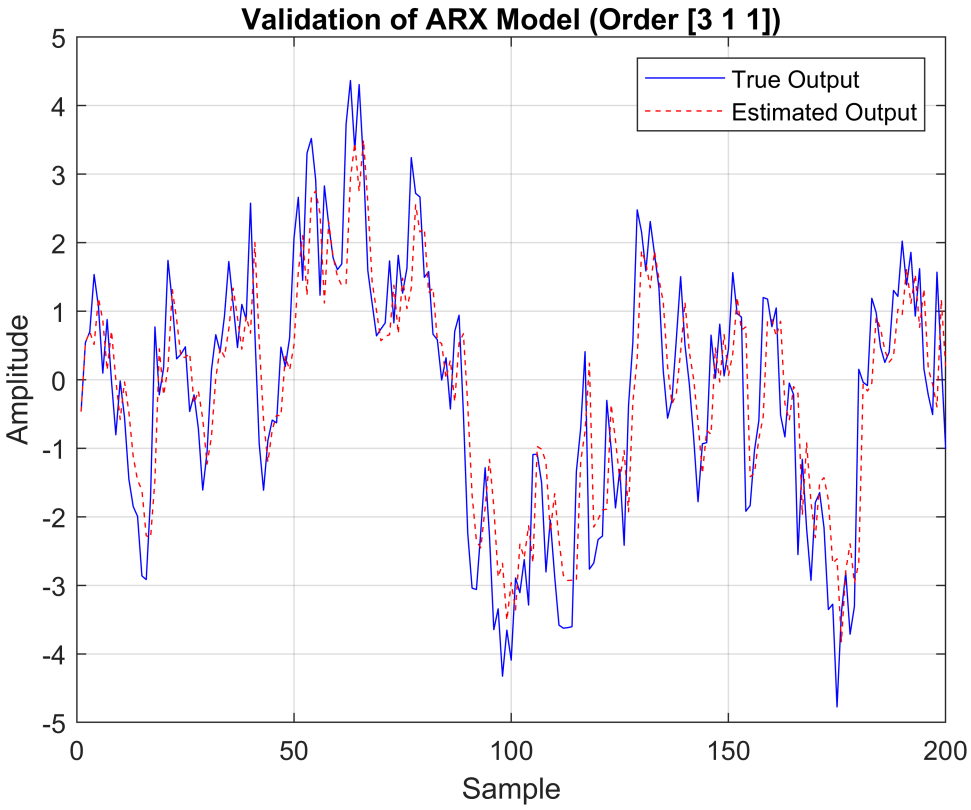
```
Estimated ARX model for order [1 1 1]:
  idpoly with properties:

               A: [1 -0.7982]
               B: [0 0.0133]
               C: 1
               D: 1
               F: 1
    IntegrateNoise: 0
          Variable: 'z^-1'
           IODelay: 0
         Structure: [1×1 pmodel.polynomial]
     NoiseVariance: 1.0246
            Report: [1×1 idresults.arx]
        InputDelay: 0
       OutputDelay: 0
                Ts: 1
          TimeUnit: 'seconds'
         InputName: {'u1'}
         InputUnit: {''}
        InputGroup: [1×1 struct]
        OutputName: {'y1'}
        OutputUnit: {''}
       OutputGroup: [1×1 struct]
             Notes: [0×1 string]
          UserData: []
              Name: ''
      SamplingGrid: [1×1 struct]
```




```
Estimated ARX model for order [2 1 1]:
  idpoly with properties:

                A: [1 -0.7667 -0.0396]
                B: [0 0.0146]
                C: 1
                D: 1
                F: 1
    IntegrateNoise: 0
          Variable: 'z^-1'
           IODelay: 0
         Structure: [1×1 pmodel.polynomial]
     NoiseVariance: 1.0255
            Report: [1×1 idresults.arx]
        InputDelay: 0
       OutputDelay: 0
                Ts: 1
          TimeUnit: 'seconds'
         InputName: {'u1'}
         InputUnit: {''}
        InputGroup: [1×1 struct]
        OutputName: {'y1'}
        OutputUnit: {''}
       OutputGroup: [1×1 struct]
             Notes: [0×1 string]
          UserData: []
              Name: ''
      SamplingGrid: [1×1 struct]
```




```
Estimated ARX model for order [3 1 1]:
  idpoly with properties:

               A: [1 -0.7648 -0.0023 -0.0488]
               B: [0 0.0147]
               C: 1
               D: 1
               F: 1
  IntegrateNoise: 0
        Variable: 'z^-1'
         IODelay: 0
       Structure: [1×1 pmodel.polynomial]
   NoiseVariance: 1.0252
          Report: [1×1 idresults.arx]
      InputDelay: 0
     OutputDelay: 0
              Ts: 1
        TimeUnit: 'seconds'
       InputName: {'u1'}
       InputUnit: {''}
      InputGroup: [1×1 struct]
      OutputName: {'y1'}
      OutputUnit: {''}
     OutputGroup: [1×1 struct]
           Notes: [0×1 string]
        UserData: []
            Name: ''
    SamplingGrid: [1×1 struct]
```


Validation of ARX Model (Order [3 1 1])
True Output
Estimated Output
Amplitude
Sample
5
4
3
2
1
0
-1
-2
-3
-4
-5
0
50
100
150
200