SE 165/265 Homework 7
Assigned May 15th, 2024; Due May 22nd, 2024 (11:59 pm Pacific Time)

Assignment Goal:
1. Extract damage sensitive features based on residual error from AR model's prediction of measured time series.
2. Develop control chart for statistical discrimination of the residual error values.
3. Reinforce material about residual errors (Section 8.2.8 of reference text), and outliers detection using control charts (Lecture 14, Section 10.5 of reference text).

Introduction
We will continue to use experimental data from the 4-story structure. You can refer to Homework #1 to recall how the data were acquired. Recall from lecture 8 that autoregressive (AR) models are a type of linear time-series model that can be fit to our measured response data.

With the 4-story structure, the first 9 states correspond to a system that is linear, but with different stiffness and mass values to simulate E&O variability one might encounter in real-world applications. We would expect the AR models that represent the data acquired from these various conditions to change as result of this variability. Damage that is introduced with the bumper mechanism in states 10-17 results in nonlinear system response somewhat analogous to crack opening and closing. Fitting AR models to data from these cases results in a model that is the best linear approximation to the data from the nonlinear system. Therefore, we would also expect the AR models to change as a result of this transition from a linear to nonlinear system, which results from the introduction of the simulated damage.

Task 1: Load data and create array

Download the file **data3SS2009.mat** from the SE 165-265 CANVAS Data File folder, 4-story structure data. **Load** this file in MATLAB. This file contains measurements from all 5 sensors.

Loading this file will give you a 3-D matrix called **dataset** ($\mathbf{8192 \times 5 \times 850}$), where 8,192 corresponds to the data points, 5 corresponds to channels 1 to 5, and 850 is the number of measurements (50 measurements each for 17 states). Recall from HW 1 that states 1-9 are considered undamaged, and states 10-17 are damaged.

In this assignment, **we will only work with data from Channel 5 (we called it sensor 4).**

Although not necessary, I first saved the data from channels 1-5 in new 8192x850 arrays called:

```
clear all
close all
load data3SS2009.mat
sensor4(1:8192,1:850)=dataset(1:8192,5,1:850);
```

For all the following tasks, we will analyze data from the **first measurement** of the following states:

1. State 1, which we consider to be the baseline.
2. State 3, a case that represents some source of variability, but no damage (a mass of 1.2kg was added to the first floor). Notice that we are NOT using State 5 this time.
3. State 10, the case with a low level of damage (hard to distinguish from undamaged cases).
4. State 12, a case with an intermediate level of damage.
5. State 14 a case with the highest level of damage.

Therefore, we will only use columns 1, 101, 451, 551, and 651.

Reminder: for all plots, add the appropriate title, labels (with the appropriate units, when there is any) and legends.

Task 1.A:
Develop 30th order Auto-Regressive (AR 30) time series model based on data from State 1 (column 1).

To do that, we will solve the least squares problem as shown in the notes from Lecture 10.

The 30th order AR model is:

$$x_i = a_1 x_{i-1} + a_2 x_{i-2} + a_3 x_{i-3} + \cdots + a_{29} x_{i-29} + a_{30} x_{i-30} \qquad (1)$$

Where $x_i = i^{th}$ acceleration value in the measured time series (in this case, the first measurement from state 1) and $a_i = AR\ coefficients$.

For a 8192 pts time series the least squares problem that solves for the coefficients a_i can be defined as:

$$\begin{bmatrix} x_1 & x_2 & x_3 & \dots & x_{30} \\ x_2 & x_3 & x_4 & \dots & x_{31} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{8162} & x_{8163} & x_{8164} & \dots & x_{8191} \end{bmatrix} \begin{Bmatrix} a_{30} \\ a_{29} \\ \vdots \\ a_1 \end{Bmatrix} = \begin{Bmatrix} x_{31} \\ x_{32} \\ \vdots \\ x_{8192} \end{Bmatrix} \qquad (2)$$

Or if we define the matrix and vectors above as $[X]\{a\} = \{B\}$.

- Create your [X] matrix.
- Create your {B} vector.
- Solve for {a} in MATLAB using the **pinv()** command as follows:

$$a = \boldsymbol{pinv}(\boldsymbol{X}) * B$$

Do not create your AR model using LPC, as it will give different results! However, you can check that you have done this right by using the MATLAB command a= lpc(X,30). The first value in this "a" vector will be one and the subsequent values should be approximately equal to the "a" values solved for using the pseudo-inverse (pinv), but with opposite signs and in opposite order (use flipud command to change the order). This is for comparison only. You should proceed with the {a} values generated with **pinv**.

Task 1-B: Generate an estimate of the measured signals (from the states specified above) using this AR model (i.e., reconstruct the time series using your AR coefficients).

This estimate is obtained by multiplying the matrix [X] above by the coefficient vector {a} that you found in the previous step (as shown in Eq. (2)).

Remember, the AR estimate of the signal will not have as many points as the actual time series because you need the first 30 time series points to form the AR estimate of 31^{st} point in the time series. Make sure you account for this difference when plotting your signals and in the subsequent calculations.

For each state (states 1, 3, 10, 12, 14), you need to create a new matrix [X] with the corresponding time series, but you will use the same AR coefficients (from state 1) for all subsequent predictions.

Therefore, we are predicting the time series responses from states 1, 3, 10, 12 & 14 using the coefficients obtained from what we consider to be the undamaged baseline data (State 1).

Task 1-C. Plot the time series:

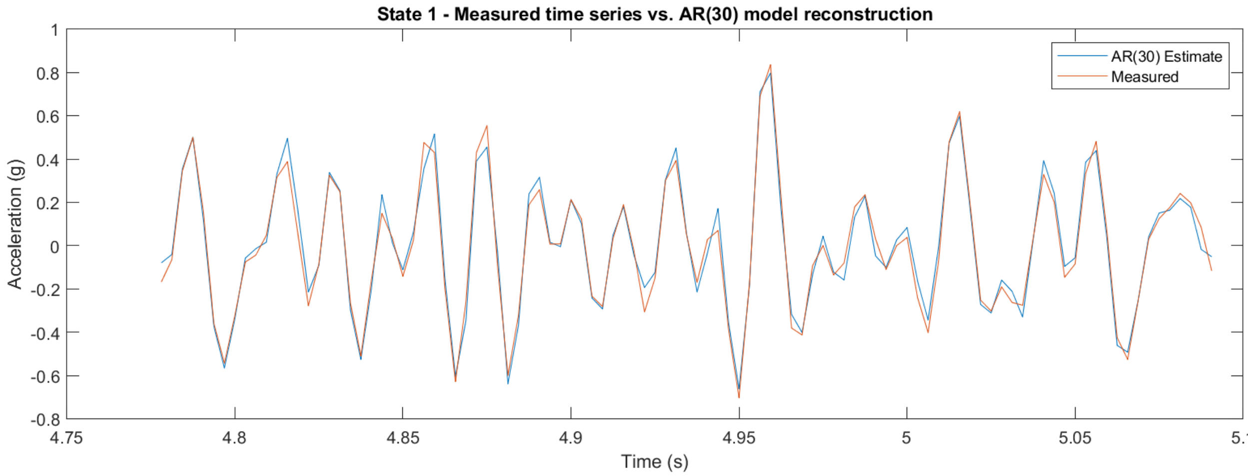
- Plot the original signal from State 1 and overlay a plot of the signal generated with AR model (they should be very similar).

  As usual, add the appropriate labels and legends.
  Again, make sure you account for the difference in the number of points in your original time series and time vector generated with the AR model.

- Plot the original signal from state 14 and overlay a plot of the signal generated with AR model. This is our highest damage level, and you will notice that the AR model doesn't reconstruct the time series very well. We expect the error to increase as the damage level increases.

Task 1-D. Repeat the plots of task 1.C, but this time zoom in a portion of these plots, points 1500 – 1600, so you can clearly see the difference between the two signals.

Hint: results from state 1.



Task 1-E. For all 5 states we are analyzing, calculate the residual error time series by subtracting time series estimated with the AR model from the measured time series.

$$\varepsilon_i = x_i - (a_1 x_{i-1} + a_2 x_{i-2} + a_3 x_{i-3} + \cdots + a_{29} x_{i-29} + a_{30} x_{i-30})$$

Again, keep in mind that your estimated time series begins at the 31^{st} point of the actual time series. Therefore, there should only be $(8192 - 30)$ residual error values. In other words, $error = measured_{31:8192} - estimated$.

Task 1-F. Residual error histograms:

- For state 1, plot a histogram of the residual errors using 60 bins. Use the MATLAB command **histfit**(e, 60) where {e} is the vector of residual errors.
- Plot another histogram for state 14, also using 60 bins.

You will be able to see how the distribution of the residual errors change from the baseline to the highest damage level.

Task 1-G. Residual error normalization:

For each state, normalize each value in the residual error time series, e_i, by subtracting the mean of the baseline residual errors from State 1 (μ_{e1}) from them, and then dividing by its standard deviation from State 1 (σ_{e1}):

$$z_i = (e_i - \mu_{e1})/\sigma_{e1}$$

where "1" corresponds to state 1, our baseline, and i corresponds to each state we are analyzing.

Task 1-H. X-bar control charts: We will develop x-bar control charts based on the average of these normalized residual error time series from task 1.G.

Use a window size (m) of 8 points. Note you can discard the last two residual errors, so your normalized error vector is divisible by 8.

For the i^{th} window, calculate the following parameters for each state:

- $\bar{x}_i(t) = mean[x(t)]$, where x(t) are the m-points normalized residual error time series contained in the i^{th} window.
- $s_i(t) = std[x(t)]$, (calculate the standard deviation of the m normalized residual error time series points contained in the i^{th} window.)

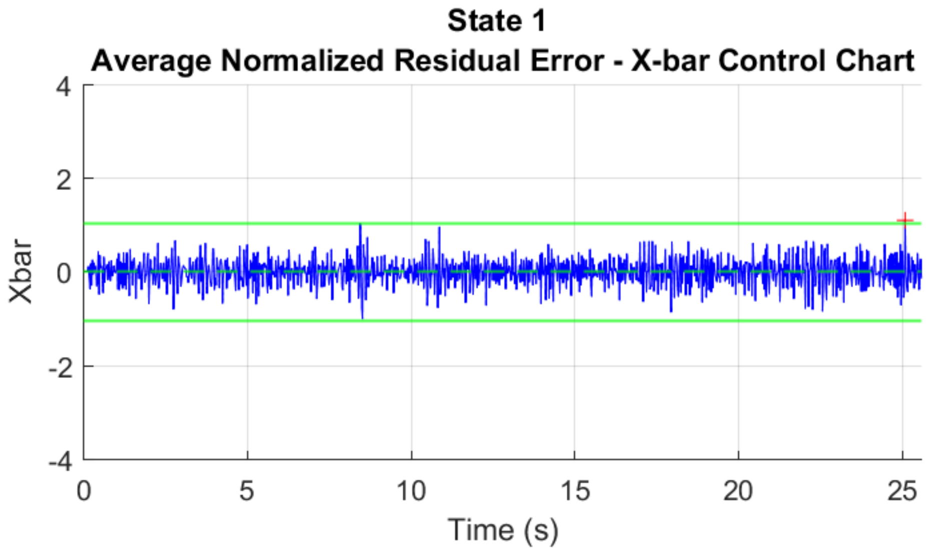
Then, calculate the control limits using the results for state 1 only. We will use the same control limits for all the states, as follows:

- $s = mean(s_i(t))$
- $CL = mean[\bar{x}_i(t)]$
- $UCL = CL + 3\frac{s}{\sqrt{m}}$, which is the upper control limit.
- $LCL = CL - 3\frac{s}{\sqrt{m}}$, which is the lower control limit.

Task 1-I. For states 1 and 14, plot histograms of $\bar{x}_i(t)$ using 30 bins (using the histfit command).

Task 1-J. Plot the control charts:

- **For all the states** (states 1, 3, 10, 12, 14), plot $\bar{x}_i(t)$, shown in blue in the hint below.
- Plot the control limits CL, UCL, and LCL as horizontal lines in green. Remember that these limits are the same for all plots (calculated from state 1).
- Flag the points that exceed the control limits, in red.
- Notice that your normalized average residual error time series has 8 times less points than the original time series. To create the new time vector, from the original time vector, I used the point at the end of the bin for each average (also accounting for the fact that our time series start at the 31^{st} point of the original time vector).
- Set y-axis limits to [-4 4] for all plots.



Hint: State 1 X-bar control chart plot.

As we observed from the previous steps, the residual errors increase as the damage level increases. We created our control chart limits using the baseline (undamaged, state 1) and we should be able to observe an increasingly number of outliers as the damage level increases.

Task 2. False positive study.

Using the AR 30 model that we developed at the beginning of task 1 (using data from State 1, measurement 1), predict the time series responses from state 1, measurement 50. **Plot the control chart for measurement 50 of state 1**.

To do that, you should repeat the steps from task 1, but we already calculated the AR(30) coefficients, so you only need to create a new [X] matrix that corresponds to measurement #50 of state 1 to reconstruct the time series.

Then, repeat the other steps: calculate the residual error, normalize it, calculate the average based on a m=8 points bin to obtain $\bar{x}_i(t)$ and plot it along with the control limits and flags for the outliers.

This problem will simulate the process of performing a false-positive study. Recall that measurement 50 and measurement 1 both correspond to state 1 with a random excitation, so there should not be many outliers.