System Identification – Practical Assignment 5
Correlation Analysis

Logistics are as before, see previous labs.

In this lab, we apply the linear regression method to obtain finite impulse response (FIR) models from input-output data – see the course material, Part IV: *Correlation Analysis*. This data is more general than just the step or impulse responses we have been seeing so far. Note that the linear regression itself will be only a tool, which you are expected to already master.

You will develop a function with the exact signature:

```
[index, ru50, ryu50, fir50, Mstar, msestar] = findfir
```

Each student is assigned an index number in the set 1-8, which needs to be saved to variable `index` at the beginning of the function. The index dictates which data file the student should load. For instance, if you have index 3, you load file `lab5_3.mat`. All these datafiles are already accessible from your function code, they have been uploaded to the Grader problem (even though they are not visible explicitly). Each file contains the identification data in variable `id`, and separately the validation data in variable `val`. Both these variables are objects of type `iddata` from the system identification toolbox of Matlab, see `doc iddata`. The corresponding time vectors are `tid`, `tval`.

Requirements follow.

- Plot and examine the data supplied. Determine whether the identification input and output are zero-mean or not. If the signals are not zero-mean, remove the means using e.g. `detrend`.
- For now, take $M = 50$. After making sure the signals are zero-mean, compute the correlation functions r_u and r_{yu} from the identification data, using the formulas from the lecture. To ensure at least $N/2$ samples are used in the correlation estimates (so that these estimates are reliable), take the maximal value of τ to be around $N/2$. Return the following column vectors: `ru50` = $[r_u(0), r_u(1), \ldots, r_u(10)]$, `ryu50` = $[r_{yu}(0), r_{yu}(1), \ldots, r_{yu}(10)]$ so that they are validated against the reference solution and you can detect any mistakes early on (remember that you cannot index by 0 in Matlab, so e.g. $r_u(0)$ will be the element at index 1 of the vector).
- Implement the system of linear equations to obtain the FIR model, solve it, and return the solution in the column vector `fir50`. Verify the model by computing the convolutions to simulate the model response to validation input, and compare with the validation outputs, both graphically and using the MSE.
- Study the influence of the length M of the FIR model on the accuracy of the model, by checking plots and MSE values for M in the set $50, 60, 70, 80, 90, 100$. Pick the value of M that leads to the smallest MSE on the validation data, and return it in `Mstar`, together with the corresponding MSE in `msestar`.
- Optionally, if you still have time, identify an FIR model of the system using `cra` with the best value of M that you found, and compute the response of this model to the validation inputs. Compare with your model. Note: the responses may not be exactly the same due to different algorithmic details, but they should look similar.

To allow for more insight, the true impulse response of the system is provided as a vector `imp` in the datafile (note that this true response would not be available in a real identification experiment). Solve each part of the assignment without using the true response, but once you have solved it, you may optionally compare the FIR models obtained with the true impulse response.

This is the first lab where we start using in earnest the System Identification toolbox (`ident`). Relevant functions from this toolbox: `cra`, `detrend`, `plot`, `compare`; other functions `conv`. See also `doc ident` for the full documentation of the toolbox.