EECS 16B Designing Information Devices and Systems II
Spring 2021 Discussion Worksheet

Discussion 7B

The relevant notes for this discussion are Note 7B and Note 8.

1. System identification by means of least squares

Working through this question will help you understand better how we can use experimental data taken from a (presumably) linear system to learn a discrete-time linear model for it using the least-squares techniques you learned in 16A. You will later do this in lab for your robot car.

As you were told in 16A, least-squares and its variants are not just the basic workhorses of machine learning in practice, they are play a conceptually central place in our understanding of machine learning well beyond least-squares.

Throughout this question, you should consider measurements to have been taken from one long trace through time.

(a) Consider the scalar discrete-time system

$$x[i+1] = ax[i] + bu[i] + w[i] \tag{1}$$

Where the scalar state at time i is $x[i]$, the input applied at time i is $u[i]$ and $w[i]$ represents some external disturbance that also participated at time i.

Assume that you have measurements for the states $x[i]$ from $i = 0$ to m and also measurements for the controls $u[i]$ from $i = 0$ to $m-1$.

Set up a least-squares problem that you can solve to get an estimate of the unknown system parameters a and b.

(b) What if there were now two distinct scalar inputs to a scalar system

$$x[i+1] = ax[i] + b_1 u_1[i] + b_2 u_2[i] + w[i] \tag{2}$$

and that we have measurements as before, but now also for both of the control inputs.

Set up a least-squares problem that you can solve to get an estimate of the unknown system parameters a, b_1, b_2.

(c) What could go wrong in the previous case? For what kind of inputs would make least-squares fail to give you the parameters you want?

(d) Returning to the scalar case with only one input, what could go wrong? When would you be unable to use least-squares to get the parameters you want?

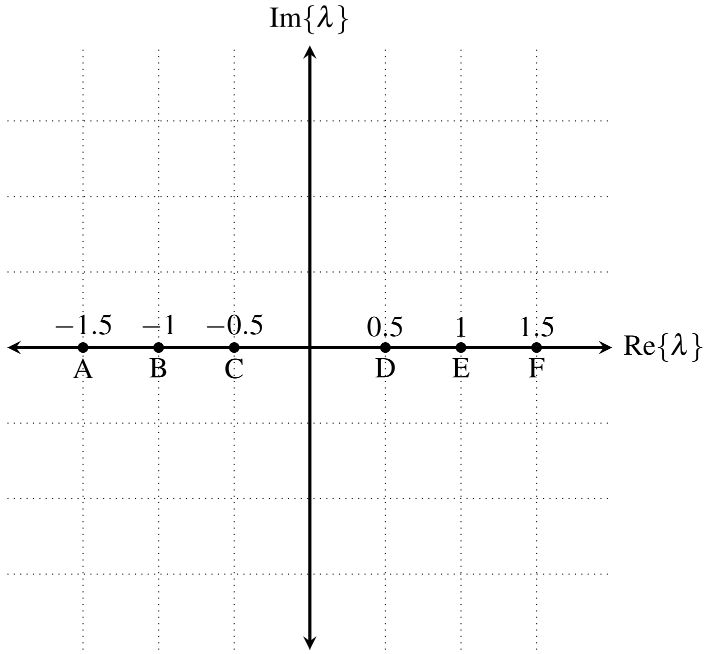
(e) Now consider the two dimensional state case with a single input.

$$\vec{x}[i+1] = \begin{bmatrix} x_1[i+1] \\ x_2[i+1] \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \vec{x}[i] + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} u[i] + \vec{w}[i] \tag{3}$$

How can we treat this like two parallel problems to set this up using least-squares to get estimates for the unknown parameters $a_{11}, a_{12}, a_{21}, a_{22}, b_1, b_2$? What work/computation can we reuse across the two problems?

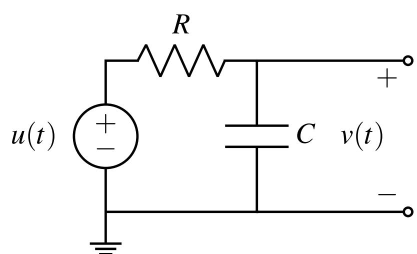
2. Discrete time system responses

We have a system $x[k+1] = \lambda x[k]$. For each λ value plotted on the real-imaginary axis, sketch $x[k]$ with an initial condition of $x[0] = 1$. Determine if each system is stable.



3. Stability Examples and Counterexamples

(a) Consider the circuit below with $R = 1\Omega, C = 0.5\text{F}$, and $u(t) = \cos(t)$. Furthermore assume that $v(0) = 0$ (that the capacitor is initially discharged).



This circuit can be modeled by the differential equation

$$\frac{d}{dt}v(t) = -2v(t) + 2u(t) \tag{4}$$

Show that the differential equation is always stable. Consider what this means in the physical circuit.

(b) Consider the discrete system

$$x[k+1] = 2x[k] + u[k] \tag{5}$$

with $x[0] = 0$.

Is the system stable or unstable? If unstable, find a bounded input sequence $u[k]$ that causes the system to “blow up”. If unstable, is there still a (non-trivial) bounded input sequence that does not cause the system to “blow up”?

Contributors:

- Anant Sahai.
- Regina Eckert.
- Kareem Ahmad.
- Kyle Tanghe.
- Sidney Buchbinder.