1 Controllability

We are given a discrete time state space system, where $\vec{x}$ is our state vector, $A$ is the state space model, $B$ is the input matrix, and $\vec{u}$ is the control input.

$$\vec{x}(t+1) = A\vec{x}(t) + B\vec{u}(t)$$  \hspace{1cm} (1)

We want to know if this system is “controllable”; if given set of inputs, we can get the system from any initial state to any final state. This has an important physical meaning; if a physical system is controllable, that means that we can get anywhere in the state space. If a robot is controllable, it is able to travel anywhere in the system it is living in (given enough control inputs).

1.1 Controllability Matrix

To figure out if a system is controllable, we can simplify the problem. If we want to reach any final state from any initial state, we can consider the initial state as the origin and the final state as any arbitrary point in the state space. A system is controllable if we start off at the initial state $\vec{x}(0) = \vec{0}$ at time $t = 0$, and after some set of control inputs $\vec{u}(t)$, we can reach an arbitrary final state $\vec{x}_0$. Let’s start the system off at $\vec{x}(0)$ and see how the system evolves with each time step.

$$\vec{x}(1) = A\vec{x}(0) + B\vec{u}(0) = A\vec{0} + B\vec{u}(0) = B\vec{u}(0)$$  \hspace{1cm} (2)

This shows us that we can go anywhere spanned by $B$ in the first time step. Using our input vector $\vec{u}$, we can push the system anywhere the matrix $B$ lets us go. Now consider the next time step.

$$\vec{x}(2) = A\vec{x}(1) + B\vec{u}(1)$$
$$= AB\vec{u}(0) + B\vec{u}(1)$$

Similarly, at this time step, we can go anywhere spanned by the columns of $[B \ AB]$. Every time step adds another degree of freedom to the system.

If we go another time step, $\vec{x}(3)$, we get the following:

$$\vec{x}(3) = A\vec{x}(2) + B\vec{u}(2)$$
$$= A^2\vec{u}(0) + A\vec{u}(1) + B\vec{u}(2)$$

After $k$ time steps, we get the following:

$$\vec{x}(k) = A\vec{x}(k-1) + B\vec{u}(k-1)$$
$$= A^{k-1}\vec{u}(0) + A^{k-2}B\vec{u}(1) + A^{k-3}B\vec{u}(2) + \cdots + AB\vec{u}(k-2) + B\vec{u}(k-1)$$
After 1 time step, we can go anywhere in the set of vectors spanned by $B$, after 2 time steps, we can go anywhere spanned by the columns of $[B \ AB]$, and after $k$ time steps, we can go anywhere spanned by the columns of the matrix $C$ defined below. This is called the “controllability” matrix.

$$C = \begin{bmatrix} B & AB & A^2B & \cdots & A^{k-2}B & A^{k-1}B \end{bmatrix}$$ (3)

If this matrix is of rank $n$ (the dimension of our state space), then our system is controllable. It means that our control system is a surjection from the domain of control inputs to the state space (i.e. that every state in the state space can be reached by at least one sequence of control inputs).

How do we know how many time steps must be checked to determine whether the system is controllable? At timestep $k+1$, we add a new vector $v_{k+1} = A^kB$ to the set of basis vectors for the state space we can reach. If $A^kB$ is linearly independent from all the previous vectors ($C_k = \begin{bmatrix} B & AB & A^2B & \cdots & A^{k-2}B & A^{k-1}B \end{bmatrix}$), then we know that adding $A^kB$ adds a dimension to the space we can reach.

However, if $A^kB$ is linearly dependent from the previous vectors $C_k$, it does not add a dimension to the space we can reach. In addition, if $A^kB$ is linearly dependent, we know that $A^{k+1}B, A^{k+2}B, \ldots$ will also never add another dimension to the space we can reach. This can be seen as follows. If $A^kB$ is linearly dependent, we can write

$$A^kB = \sum_{j=0}^{k-1} \alpha_j A^jB$$

If we consider $A^{k+1}B$, we can write

$$A^{k+1}B = AA^kB = A \left( \sum_{j=0}^{k-1} \alpha_j A^jB \right) = A \left( \sum_{j=0}^{k-2} \alpha_j A^jB + \alpha_{k-1} A^{k-1}B \right) = \sum_{j=0}^{k-2} \alpha_j A^{j+1}B + \alpha_{k-1} A^kB = \sum_{j=0}^{k-2} \alpha_j A^{j+1}B + \alpha_{k-1} \left( \sum_{j=0}^{k-1} \alpha_j A^jB \right)$$

As this is just a sum of the terms in $C_k$, we see that $A^{k+1}B$ is also linearly dependent.

Finally, $A$ is an $n \times n$ matrix, there are at most $n$ dimensions in the state space, so we need to only check up to the $n^{th}$ timestep (which corresponds to $A^{n-1}B$).

Putting this all together, we get

$$C = \begin{bmatrix} B & AB & A^2B & \cdots & A^{n-2}B & A^{n-1}B \end{bmatrix}$$ (4)

Given a discrete time system $\bar{x}$ of dimension $n$, the system is controllable if its controllability matrix $C$ is of rank $n$. If a system is controllable, then given a starting position $\bar{x}(0) = \bar{0}$, it takes a maximum of $n$ control inputs over $n$ time steps for the system to reach any final state $\bar{x}_0$.

### 1. Uncontrollability
Consider the following discrete-time system with the given initial state:

\[
\vec{x}(t + 1) = \begin{bmatrix} 2 & 0 & 0 \\ -3 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \vec{x}(t) + \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix} u(t) \\
\vec{x}(0) = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}
\]

(a) Is the system controllable?

(b) Is it possible to reach \(\vec{x}(T) = \begin{bmatrix} 2 \\ -3 \\ -2 \end{bmatrix}\) for some \(t = T\)? For what input sequence \(u(t)\) up to \(t = T - 1\)?

(c) Find the set of all possible states reachable after two timesteps.

(d) Is it possible to reach \(\vec{x}(T) = \begin{bmatrix} -2 \\ 4 \\ 6 \end{bmatrix}\) for some \(t = T\)? For what input sequence \(u(t)\) up to \(t = T - 1\)?

2. 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

\[
x(i + 1) = ax(i) + bu(i) + w(i)
\]

(5)

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

\[
x(i + 1) = ax(i) + b_{1,1}u_1(i) + b_{1,2}u_2(i) + w(i)
\]

(6)

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** \(a, b_{1,1}, b_{1,2}\).
(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.

\[
\vec{x}(i+1) = \begin{bmatrix} x_1(i+1) \\ x_2(i+1) \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \vec{x}(i) + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} u_1(i) + \vec{w}(i)
\]

(7)

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?

Contributors:

- Saavan Patel.
- Sidney Buchbinder.
- Titan Yuan.
- Anant Sahai.
- Regina Eckert.