Lecture 1
Overview

- course mechanics
- outline & topics
- what is a linear dynamical system?
- why study linear systems?
- some examples

Course mechanics

- all class info, lectures, homeworks, announcements on class web page:

`www.stanford.edu/class/ee263`

course requirements:

- weekly homework
- takehome midterm exam (date TBD)
- takehome final exam (date TBD)

Prerequisites

- exposure to linear algebra (*e.g.*, Math 103)
- exposure to Laplace transform, differential equations

not needed, but might increase appreciation:

- control systems
- circuits & systems
- dynamics

Major topics & outline

- linear algebra & applications
- autonomous linear dynamical systems
- linear dynamical systems with inputs & outputs
- basic quadratic control & estimation

Linear dynamical system

continuous-time linear dynamical system (CT LDS) has the form

$$\frac{dx}{dt} = A(t)x(t) + B(t)u(t), \quad y(t) = C(t)x(t) + D(t)u(t)$$

where:

- $t \in \mathbf{R}$ denotes *time*
- $x(t) \in \mathbf{R}^n$ is the *state* (vector)
- $u(t) \in \mathbf{R}^m$ is the *input* or *control*
- $y(t) \in \mathbf{R}^p$ is the *output*

- $A(t) \in \mathbf{R}^{n\times n}$ is the *dynamics matrix*
- $B(t) \in \mathbf{R}^{n\times m}$ is the *input matrix*
- $C(t) \in \mathbf{R}^{p\times n}$ is the *output* or *sensor matrix*
- $D(t) \in \mathbf{R}^{p\times m}$ is the *feedthrough matrix*

for lighter appearance, equations are often written

$$\dot{x} = Ax + Bu, \qquad y = Cx + Du$$

- CT LDS is a first order vector *differential equation*
- also called *state equations*, or 'm-input, n-state, p-output' LDS

Some LDS terminology

- most linear systems encountered are *time-invariant*: A, B, C, D are constant, *i.e.*, don't depend on t
- when there is no input u (hence, no B or D) system is called *autonomous*
- very often there is no feedthrough, *i.e.*, $D = 0$
- when $u(t)$ and $y(t)$ are scalar, system is called *single-input, single-output* (SISO); when input & output signal dimensions are more than one, MIMO

Discrete-time linear dynamical system

discrete-time linear dynamical system (DT LDS) has the form

$$x(t+1) = A(t)x(t) + B(t)u(t), \qquad y(t) = C(t)x(t) + D(t)u(t)$$

where

- $t \in \mathbf{Z} = \{0, \pm 1, \pm 2, \ldots\}$
- (vector) signals x, u, y are *sequences*

DT LDS is a first order vector *recursion*

Why study linear systems?

applications arise in **many** areas, *e.g.*

- automatic control systems
- signal processing
- communications
- economics, finance
- circuit analysis, simulation, design
- mechanical and civil engineering
- aeronautics
- navigation, guidance

Usefulness of LDS

- depends on availability of **computing power**, which is large & increasing exponentially
- used for
  - analysis & design
  - implementation, embedded in real-time systems
- like DSP, was a specialized topic & technology 30 years ago

Origins and history

- parts of LDS theory can be traced to 19th century
- builds on classical circuits & systems (1920s on) (transfer functions . . .) but with more emphasis on linear algebra
- first engineering application: aerospace, 1960s
- transitioned from specialized topic to ubiquitous in 1980s (just like digital signal processing, information theory, . . .)

Nonlinear dynamical systems

many dynamical systems are **nonlinear** (a fascinating topic) so why study **linear** systems?

- most techniques for nonlinear systems are based on linear methods
- methods for linear systems often work unreasonably well, in practice, for nonlinear systems
- if you don't understand linear dynamical systems you certainly can't understand nonlinear dynamical systems

Examples (ideas only, no details)

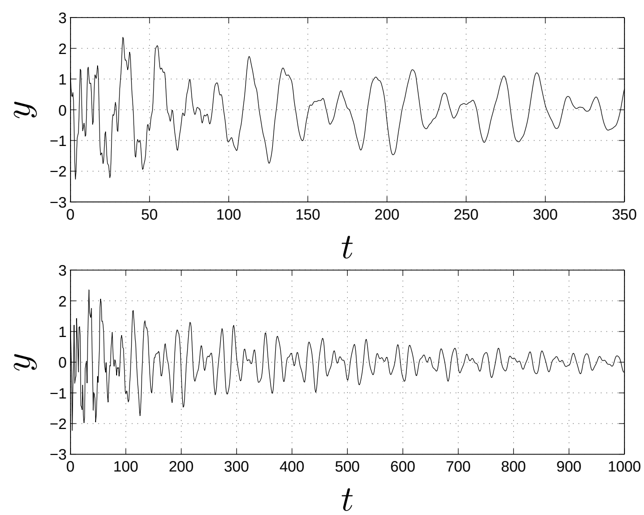
- let's consider a specific system

$$\dot{x} = Ax, \qquad y = Cx$$

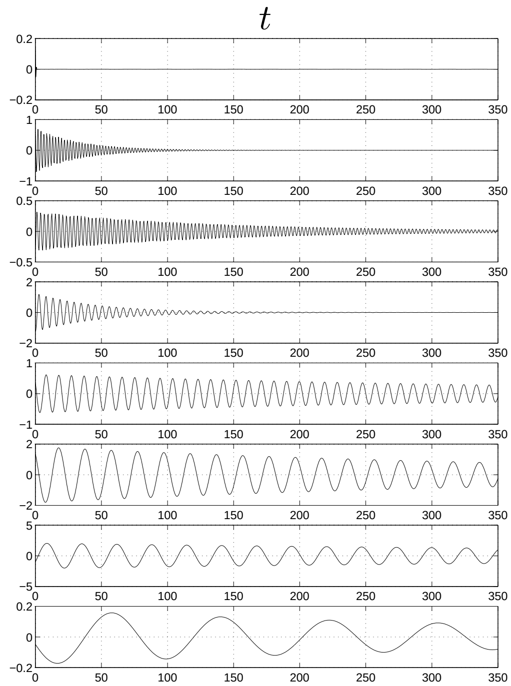
  with $x(t) \in \mathbf{R}^{16}$, $y(t) \in \mathbf{R}$ (a '16-state single-output system')

- model of a lightly damped mechanical system, but it doesn't matter

typical output:



- output waveform is very complicated; looks almost random and unpredictable
- we'll see that such a solution can be decomposed into much simpler (modal) components



(idea probably familiar from 'poles')

Input design

add two inputs, two outputs to system:

$$\dot{x} = Ax + Bu, \quad y = Cx, \quad x(0) = 0$$

where $B \in \mathbf{R}^{16\times 2}$, $C \in \mathbf{R}^{2\times 16}$ (same A as before)

problem: find appropriate $u : \mathbf{R}_+ \to \mathbf{R}^2$ so that $y(t) \to y_{\rm des} = (1, -2)$

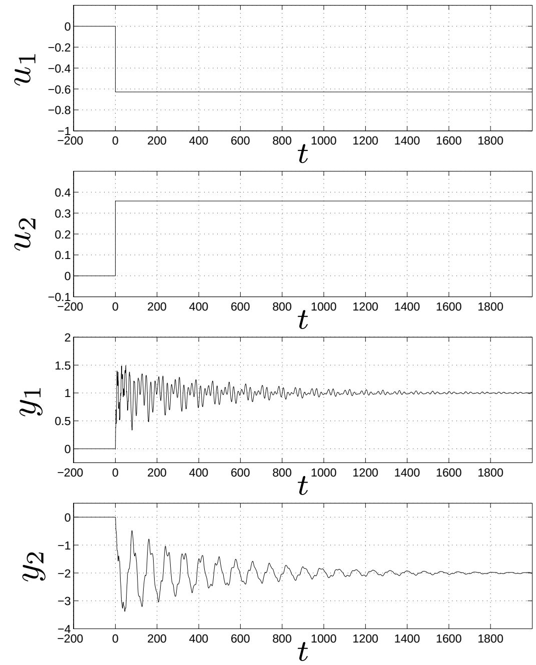
simple approach: consider static conditions (u, x, y constant):

$$\dot{x} = 0 = Ax + Bu_{\rm static}, \quad y = y_{\rm des} = Cx$$

solve for u to get:

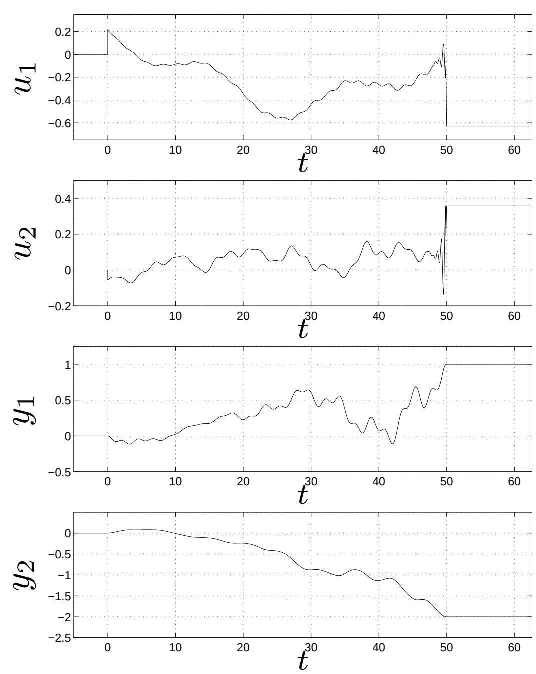
$$u_{\rm static} = \left(-CA^{-1}B\right)^{-1} y_{\rm des} = \begin{bmatrix} -0.63 \\ 0.36 \end{bmatrix}$$

let's apply $u = u_{\text{static}}$ and just wait for things to settle:

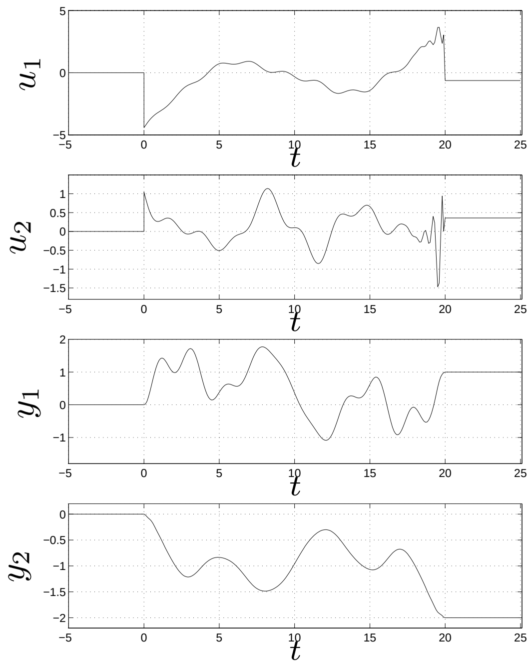


. . . takes about 1500 sec for $y(t)$ to converge to y_{des}

using very clever input waveforms (EE263) we can do much better, *e.g.*



. . . here y converges *exactly* in 50 sec in fact by using larger inputs we do still better, $e.g.$

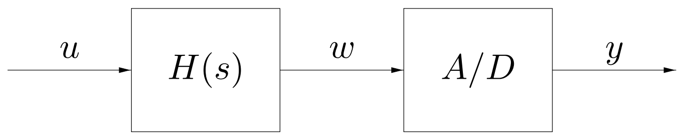


. . . here we have (exact) convergence in 20 sec

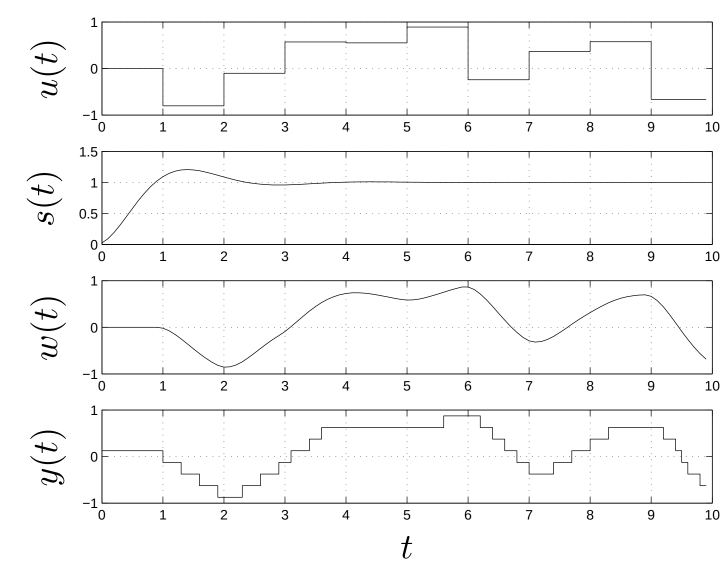
in this course we'll study

- how to synthesize or design such inputs
- the tradeoff between size of u and convergence time

Estimation / filtering



- signal u is piecewise constant (period $1\,\mathrm{sec}$)
- filtered by 2nd-order system $H(s)$, step response $s(t)$
- A/D runs at 10Hz, with 3-bit quantizer



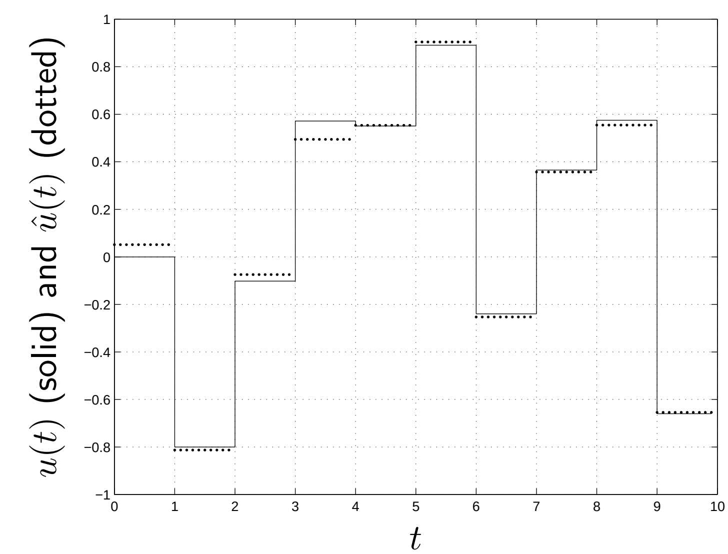
problem: estimate original signal u, given quantized, filtered signal y

simple approach:

- ignore quantization
- design equalizer $G(s)$ for $H(s)$ ($i.e.$, $GH \approx 1$)
- approximate u as $G(s)y$

. . . yields terrible results

formulate as *estimation problem* (EE263) . . .



RMS error 0.03, well **below** quantization error (!)

Lecture 2
Linear functions and examples

- linear equations and functions
- engineering examples
- interpretations

Linear equations

consider system of linear equations

$$\begin{array}{rcccccccc} y_1 & = & a_{11}x_1 & + & a_{12}x_2 & + \cdots + & a_{1n}x_n \\ y_2 & = & a_{21}x_1 & + & a_{22}x_2 & + \cdots + & a_{2n}x_n \\ & \vdots & & & & & \\ y_m & = & a_{m1}x_1 & + & a_{m2}x_2 & + \cdots + & a_{mn}x_n \end{array}$$

can be written in matrix form as $y = Ax$, where

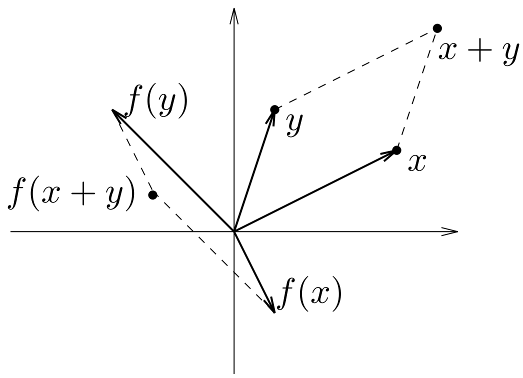
$$y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} \quad A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \quad x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

Linear functions

a function $f : \mathbf{R}^n \longrightarrow \mathbf{R}^m$ is *linear* if

- $f(x + y) = f(x) + f(y)$, $\forall x, y \in \mathbf{R}^n$
- $f(\alpha x) = \alpha f(x)$, $\forall x \in \mathbf{R}^n$ $\forall \alpha \in \mathbf{R}$

i.e., *superposition* holds



Matrix multiplication function

- consider function $f : \mathbf{R}^n \to \mathbf{R}^m$ given by $f(x) = Ax$, where $A \in \mathbf{R}^{m \times n}$
- matrix multiplication function f is linear
- **converse** is true: **any** linear function $f : \mathbf{R}^n \to \mathbf{R}^m$ can be written as $f(x) = Ax$ for some $A \in \mathbf{R}^{m \times n}$
- representation via matrix multiplication is unique: for any linear function f there is only one matrix A for which $f(x) = Ax$ for all x
- $y = Ax$ is a concrete representation of a generic linear function

Interpretations of $y = Ax$

- y is measurement or observation; x is unknown to be determined
- x is 'input' or 'action'; y is 'output' or 'result'
- $y = Ax$ defines a function or transformation that maps $x \in \mathbf{R}^n$ into $y \in \mathbf{R}^m$

Interpretation of a_{ij}

$$y_i = \sum_{j=1}^{n} a_{ij} x_j$$

a_{ij} is *gain factor* from jth input (x_j) to ith output (y_i)

thus, $e.g.$,

- ith *row* of A concerns ith *output*
- jth *column* of A concerns jth *input*
- $a_{27} = 0$ means 2nd output (y_2) doesn't depend on 7th input (x_7)
- $|a_{31}| \gg |a_{3j}|$ for $j \neq 1$ means y_3 depends mainly on x_1

- $|a_{52}| \gg |a_{i2}|$ for $i \neq 5$ means x_2 affects mainly y_5

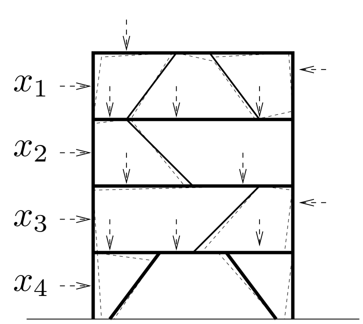
- A is lower triangular, *i.e.*, $a_{ij} = 0$ for $i < j$, means y_i only depends on $x_1, \ldots, x_i$

- A is diagonal, *i.e.*, $a_{ij} = 0$ for $i \neq j$, means ith output depends only on ith input

more generally, **sparsity pattern** of A, *i.e.*, list of zero/nonzero entries of A, shows which x_j affect which y_i

Linear elastic structure

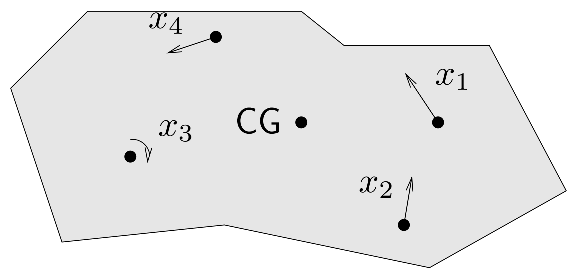
- x_j is external force applied at some node, in some fixed direction
- y_i is (small) deflection of some node, in some fixed direction



(provided x, y are small) we have $y \approx Ax$

- A is called the *compliance matrix*
- a_{ij} gives deflection i per unit force at j (in m/N)

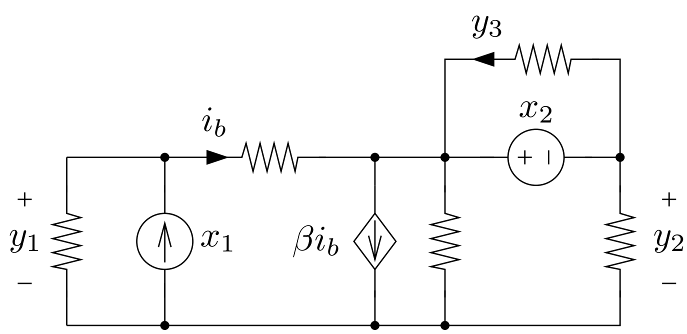
Total force/torque on rigid body



- x_j is external force/torque applied at some point/direction/axis
- $y \in \mathbf{R}^6$ is resulting total force & torque on body
  ($y_1,\ y_2,\ y_3$ are $\mathbf{x}$-, $\mathbf{y}$-, $\mathbf{z}$- components of total force,
  $y_4,\ y_5,\ y_6$ are $\mathbf{x}$-, $\mathbf{y}$-, $\mathbf{z}$- components of total torque)
- we have $y = Ax$
- A depends on geometry
  (of applied forces and torques with respect to center of gravity CG)
- jth column gives resulting force & torque for unit force/torque j

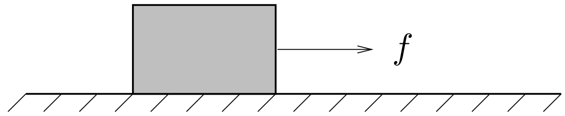
Linear static circuit

interconnection of resistors, linear dependent (controlled) sources, and independent sources



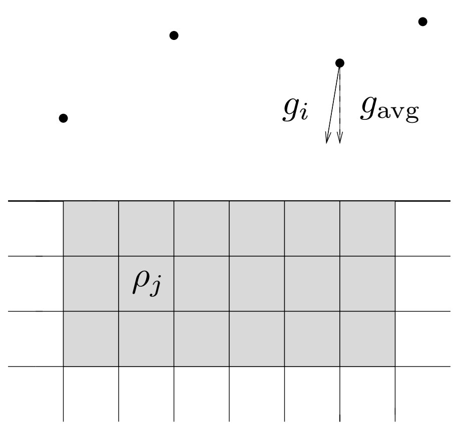
- x_j is value of independent source j
- y_i is some circuit variable (voltage, current)
- we have $y = Ax$
- if x_j are currents and y_i are voltages, A is called the *impedance* or *resistance* matrix

Final position/velocity of mass due to applied forces



- unit mass, zero position/velocity at $t = 0$, subject to force $f(t)$ for $0 \leq t \leq n$
- $f(t) = x_j$ for $j - 1 \leq t < j$, $j = 1, \ldots, n$
  (x is the sequence of applied forces, constant in each interval)
- y_1, y_2 are final position and velocity ($i.e.$, at $t = n$)
- we have $y = Ax$
- a_{1j} gives influence of applied force during $j - 1 \leq t < j$ on final position
- a_{2j} gives influence of applied force during $j - 1 \leq t < j$ on final velocity

Gravimeter prospecting



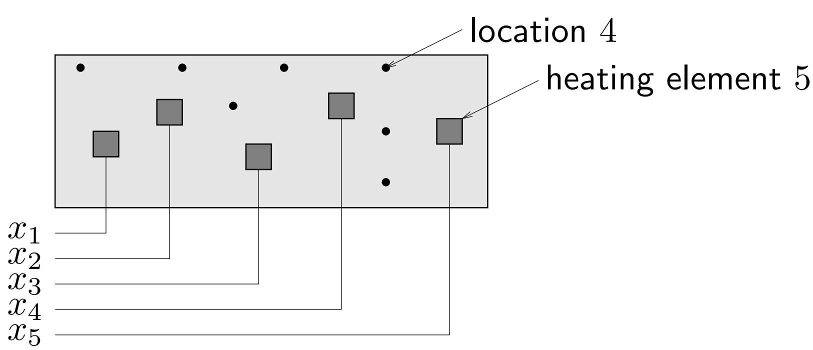
- $x_j = \rho_j - \rho_{\text{avg}}$ is (excess) mass density of earth in voxel j;
- y_i is measured *gravity anomaly* at location i, $i.e.$, some component (typically vertical) of $g_i - g_{\text{avg}}$
- $y = Ax$

- A comes from physics and geometry

- jth column of A shows sensor readings caused by unit density anomaly at voxel j

- ith row of A shows sensitivity pattern of sensor i

Thermal system



- x_j is power of jth heating element or heat source

- y_i is change in steady-state temperature at location i

- thermal transport via conduction

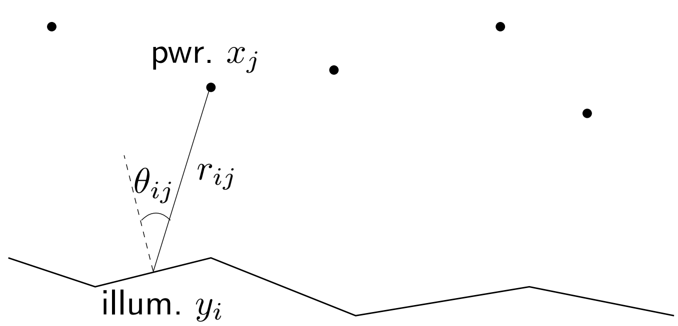
- $y = Ax$

- a_{ij} gives influence of heater j at location i (in °C/W)

- jth column of A gives pattern of steady-state temperature rise due to 1W at heater j

- ith row shows how heaters affect location i

Illumination with multiple lamps



- n lamps illuminating m (small, flat) patches, no shadows
- x_j is power of jth lamp; y_i is illumination level of patch i
- $y = Ax$, where $a_{ij} = r_{ij}^{-2} \max\{\cos\theta_{ij}, 0\}$
  ($\cos\theta_{ij} < 0$ means patch i is shaded from lamp j)
- jth column of A shows illumination pattern from lamp j

Signal and interference power in wireless system

- n transmitter/receiver pairs
- transmitter j transmits to receiver j (and, inadvertantly, to the other receivers)
- p_j is power of jth transmitter
- s_i is received signal power of ith receiver
- z_i is received interference power of ith receiver
- G_{ij} is path gain from transmitter j to receiver i
- we have $s = Ap$, $z = Bp$, where

$$a_{ij} = \begin{cases} G_{ii} & i = j \\ 0 & i \neq j \end{cases} \qquad b_{ij} = \begin{cases} 0 & i = j \\ G_{ij} & i \neq j \end{cases}$$

- A is diagonal; B has zero diagonal (ideally, A is 'large', B is 'small')

Cost of production

production *inputs* (materials, parts, labor, . . .) are combined to make a number of *products*

- x_j is price per unit of production input j
- a_{ij} is units of production input j required to manufacture one unit of product i
- y_i is production cost per unit of product i
- we have $y = Ax$
- ith row of A is *bill of materials* for unit of product i

production inputs needed

- q_i is quantity of product i to be produced
- r_j is total quantity of production input j needed
- we have $r = A^T q$

total production cost is

$$r^T x = (A^T q)^T x = q^T A x$$

Network traffic and flows

- n flows with rates $f_1, \ldots, f_n$ pass from their source nodes to their destination nodes over fixed routes in a network
- t_i, traffic on link i, is sum of rates of flows passing through it
- flow routes given by *flow-link incidence matrix*

$$A_{ij} = \begin{cases} 1 & \text{flow } j \text{ goes over link } i \\ 0 & \text{otherwise} \end{cases}$$

- traffic and flow rates related by $t = Af$

link delays and flow latency

- let $d_1, \ldots, d_m$ be link delays, and $l_1, \ldots, l_n$ be latency (total travel time) of flows
- $l = A^T d$
- $f^T l = f^T A^T d = (Af)^T d = t^T d$, total # of packets in network

Linearization

- if $f : \mathbf{R}^n \to \mathbf{R}^m$ is differentiable at $x_0 \in \mathbf{R}^n$, then

$$x \text{ near } x_0 \Longrightarrow f(x) \text{ very near } f(x_0) + Df(x_0)(x - x_0)$$

  where

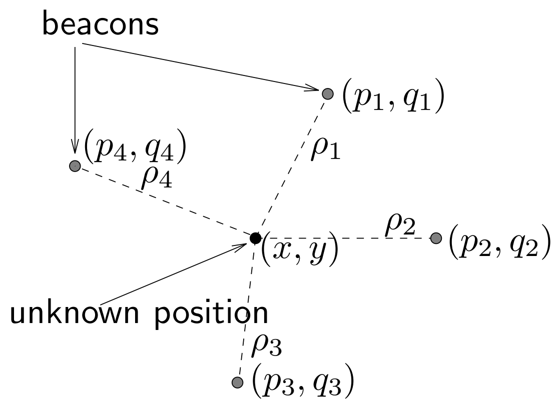
$$Df(x_0)_{ij} = \left.\frac{\partial f_i}{\partial x_j}\right|_{x_0}$$

  is derivative (Jacobian) matrix

- with $y = f(x)$, $y_0 = f(x_0)$, define *input deviation* $\delta x := x - x_0$, *output deviation* $\delta y := y - y_0$
- then we have $\delta y \approx Df(x_0)\delta x$
- when deviations are small, they are (approximately) related by a linear function

Navigation by range measurement

- (x, y) unknown coordinates in plane
- (p_i, q_i) known coordinates of beacons for $i = 1, 2, 3, 4$
- ρ_i measured (known) distance or range from beacon i



- $\rho \in \mathbf{R}^4$ is a nonlinear function of $(x, y) \in \mathbf{R}^2$:

$$\rho_i(x, y) = \sqrt{(x - p_i)^2 + (y - q_i)^2}$$

- linearize around (x_0, y_0): $\delta\rho \approx A \begin{bmatrix} \delta x \\ \delta y \end{bmatrix}$, where

$$a_{i1} = \frac{(x_0 - p_i)}{\sqrt{(x_0 - p_i)^2 + (y_0 - q_i)^2}}, \quad a_{i2} = \frac{(y_0 - q_i)}{\sqrt{(x_0 - p_i)^2 + (y_0 - q_i)^2}}$$

- ith row of A shows (approximate) change in ith range measurement for (small) shift in (x, y) from (x_0, y_0)
- first column of A shows sensitivity of range measurements to (small) change in x from x_0
- obvious application: (x_0, y_0) is last navigation fix; (x, y) is current position, a short time later

Broad categories of applications

linear model or function $y = Ax$

some broad categories of applications:

- estimation or inversion
- control or design
- mapping or transformation

(this list is not exclusive; can have combinations . . .)

Estimation or inversion

$$y = Ax$$

- y_i is ith measurement or sensor reading (which we know)
- x_j is jth parameter to be estimated or determined
- a_{ij} is sensitivity of ith sensor to jth parameter

sample problems:

- find x, given y
- find all x's that result in y ($i.e.$, all x's consistent with measurements)
- if there is no x such that $y = Ax$, find x s.t. $y \approx Ax$ ($i.e.$, if the sensor readings are inconsistent, find x which is almost consistent)

Control or design

$$y = Ax$$

- x is vector of design parameters or inputs (which we can choose)
- y is vector of results, or outcomes
- A describes how input choices affect results

sample problems:

- find x so that $y = y_{\rm des}$
- find all x's that result in $y = y_{\rm des}$ ($i.e.$, find all designs that meet specifications)
- among x's that satisfy $y = y_{\rm des}$, find a small one ($i.e.$, find a small or efficient x that meets specifications)

Mapping or transformation

- x is mapped or transformed to y by linear function $y = Ax$

sample problems:

- determine if there is an x that maps to a given y
- (if possible) find *an* x that maps to y
- find *all* x's that map to a given y
- if there is only one x that maps to y, find it ($i.e.$, decode or undo the mapping)

Matrix multiplication as mixture of columns

write $A \in \mathbf{R}^{m \times n}$ in terms of its columns:

$$A = \left[\begin{array}{cccc} a_1 & a_2 & \cdots & a_n \end{array}\right]$$

where $a_j \in \mathbf{R}^m$

then $y = Ax$ can be written as

$$y = x_1 a_1 + x_2 a_2 + \cdots + x_n a_n$$

(x_j's are scalars, a_j's are m-vectors)

- y is a (linear) combination or mixture of the columns of A
- coefficients of x give coefficients of mixture

an important example: $x = e_j$, the jth *unit vector*

$$e_1 = \left[\begin{array}{c} 1 \\ 0 \\ \vdots \\ 0 \end{array}\right], \quad e_2 = \left[\begin{array}{c} 0 \\ 1 \\ \vdots \\ 0 \end{array}\right], \quad \ldots \quad e_n = \left[\begin{array}{c} 0 \\ 0 \\ \vdots \\ 1 \end{array}\right]$$

then $Ae_j = a_j$, the jth column of A

(e_j corresponds to a pure mixture, giving only column j)

Matrix multiplication as inner product with rows

write A in terms of its rows:

$$A = \begin{bmatrix} \tilde{a}_1^T \\ \tilde{a}_2^T \\ \vdots \\ \tilde{a}_n^T \end{bmatrix}$$

where $\tilde{a}_i \in \mathbf{R}^n$

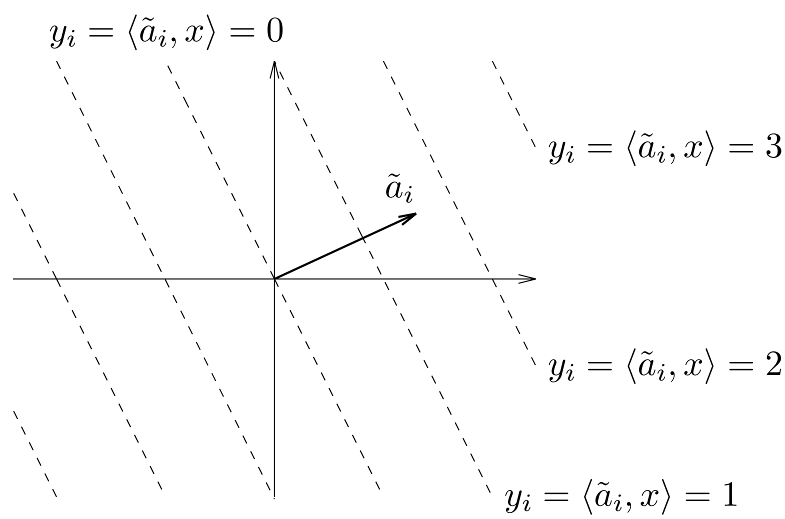
then $y = Ax$ can be written as

$$y = \begin{bmatrix} \tilde{a}_1^T x \\ \tilde{a}_2^T x \\ \vdots \\ \tilde{a}_m^T x \end{bmatrix}$$

thus $y_i = \langle \tilde{a}_i, x \rangle$, $i.e.$, y_i is inner product of ith row of A with x

geometric interpretation:

$y_i = \tilde{a}_i^T x = \alpha$ is a hyperplane in $\mathbf{R}^n$ (normal to $\tilde{a}_i$)



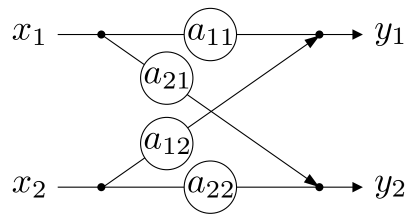
Block diagram representation

$y = Ax$ can be represented by a *signal flow graph* or *block diagram*

$e.g.$ for $m = n = 2$, we represent

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

as



- a_{ij} is the gain along the path from jth input to ith output
- (by not drawing paths with zero gain) shows sparsity structure of A ($e.g.$, diagonal, block upper triangular, arrow . . .)

example: block upper triangular, $i.e.$,

$$A = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix}$$

where $A_{11} \in \mathbf{R}^{m_1 \times n_1}$, $A_{12} \in \mathbf{R}^{m_1 \times n_2}$, $A_{21} \in \mathbf{R}^{m_2 \times n_1}$, $A_{22} \in \mathbf{R}^{m_2 \times n_2}$

partition x and y conformably as

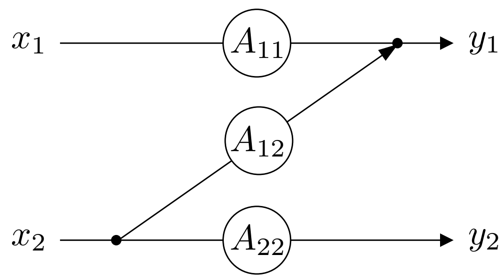
$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \qquad y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$$

($x_1 \in \mathbf{R}^{n_1}$, $x_2 \in \mathbf{R}^{n_2}$, $y_1 \in \mathbf{R}^{m_1}$, $y_2 \in \mathbf{R}^{m_2}$) so

$$y_1 = A_{11}x_1 + A_{12}x_2, \qquad y_2 = A_{22}x_2,$$

$i.e.$, y_2 doesn't depend on x_1

block diagram:



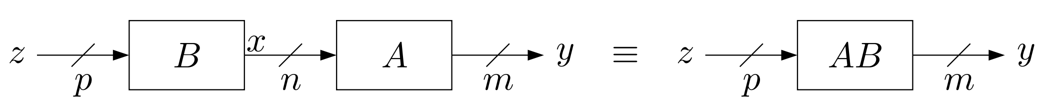
... no path from x_1 to y_2, so y_2 doesn't depend on x_1

Matrix multiplication as composition

for $A \in \mathbf{R}^{m\times n}$ and $B \in \mathbf{R}^{n\times p}$, $C = AB \in \mathbf{R}^{m\times p}$ where

$$c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$$

composition interpretation: $y = Cz$ represents composition of $y = Ax$ and $x = Bz$



(note that B is on left in block diagram)

Column and row interpretations

can write product $C = AB$ as

$$C = \left[\begin{array}{ccc} c_1 & \cdots & c_p \end{array} \right] = AB = \left[\begin{array}{ccc} Ab_1 & \cdots & Ab_p \end{array} \right]$$

$i.e.$, ith column of C is A acting on ith column of B

similarly we can write

$$C = \left[\begin{array}{c} \tilde{c}_1^T \\ \vdots \\ \tilde{c}_m^T \end{array} \right] = AB = \left[\begin{array}{c} \tilde{a}_1^T B \\ \vdots \\ \tilde{a}_m^T B \end{array} \right]$$

$i.e.$, ith row of C is ith row of A acting (on left) on B

Inner product interpretation

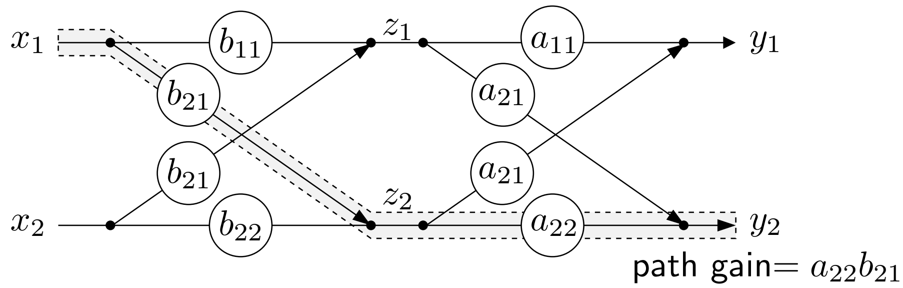
inner product interpretation:

$$c_{ij} = \tilde{a}_i^T b_j = \langle \tilde{a}_i, b_j \rangle$$

$i.e.$, entries of C are inner products of rows of A and columns of B

- $c_{ij} = 0$ means ith row of A is orthogonal to jth column of B
- **Gram matrix** of vectors $f_1, \ldots, f_n$ defined as $G_{ij} = f_i^T f_j$ (gives inner product of each vector with the others)
- $G = [f_1 \ \cdots \ f_n]^T [f_1 \ \cdots \ f_n]$

Matrix multiplication interpretation via paths



- $a_{ik}b_{kj}$ is gain of path from input j to output i via k
- c_{ij} is sum of gains over *all* paths from input j to output i

Lecture 3
Linear algebra review

- vector space, subspaces
- independence, basis, dimension
- range, nullspace, rank
- change of coordinates
- norm, angle, inner product

Vector spaces

a *vector space* or *linear space* (over the reals) consists of

- a set $\mathcal{V}$
- a vector sum $+ : \mathcal{V} \times \mathcal{V} \to \mathcal{V}$
- a scalar multiplication : $\mathbf{R} \times \mathcal{V} \to \mathcal{V}$
- a distinguished element $0 \in \mathcal{V}$

which satisfy a list of properties

- $x + y = y + x, \quad \forall x, y \in \mathcal{V}$ (+ is commutative)
- $(x + y) + z = x + (y + z), \quad \forall x, y, z \in \mathcal{V}$ (+ is associative)
- $0 + x = x$, $\forall x \in \mathcal{V}$ (0 is additive identity)
- $\forall x \in \mathcal{V} \;\; \exists (-x) \in \mathcal{V}$ s.t. $x + (-x) = 0$ (existence of additive inverse)
- $(\alpha\beta)x = \alpha(\beta x), \quad \forall \alpha, \beta \in \mathbf{R} \quad \forall x \in \mathcal{V}$ (scalar mult. is associative)
- $\alpha(x + y) = \alpha x + \alpha y, \quad \forall \alpha \in \mathbf{R} \;\; \forall x, y \in \mathcal{V}$ (right distributive rule)
- $(\alpha + \beta)x = \alpha x + \beta x, \quad \forall \alpha, \beta \in \mathbf{R} \quad \forall x \in \mathcal{V}$ (left distributive rule)
- $1x = x, \quad \forall x \in \mathcal{V}$

Examples

- $\mathcal{V}_1 = \mathbf{R}^n$, with standard (componentwise) vector addition and scalar multiplication
- $\mathcal{V}_2 = \{0\}$ (where $0 \in \mathbf{R}^n$)
- $\mathcal{V}_3 = \mathrm{span}(v_1, v_2, \ldots, v_k)$ where

$$\mathrm{span}(v_1, v_2, \ldots, v_k) = \{\alpha_1 v_1 + \cdots + \alpha_k v_k \mid \alpha_i \in \mathbf{R}\}$$

and $v_1, \ldots, v_k \in \mathbf{R}^n$

Subspaces

- a *subspace* of a vector space is a *subset* of a vector space which is itself a vector space
- roughly speaking, a subspace is closed under vector addition and scalar multiplication
- examples $\mathcal{V}_1$, $\mathcal{V}_2$, $\mathcal{V}_3$ above are subspaces of $\mathbf{R}^n$

Vector spaces of functions

- $\mathcal{V}_4 = \{x : \mathbf{R}_+ \to \mathbf{R}^n \mid x \text{ is differentiable}\}$, where vector sum is sum of functions:
$$(x + z)(t) = x(t) + z(t)$$
and scalar multiplication is defined by
$$(\alpha x)(t) = \alpha x(t)$$
(a *point* in $\mathcal{V}_4$ is a *trajectory* in $\mathbf{R}^n$)
- $\mathcal{V}_5 = \{x \in \mathcal{V}_4 \mid \dot{x} = Ax\}$
(*points* in $\mathcal{V}_5$ are *trajectories* of the linear system $\dot{x} = Ax$)
- $\mathcal{V}_5$ is a subspace of $\mathcal{V}_4$

Independent set of vectors

a set of vectors $\{v_1, v_2, \ldots, v_k\}$ is *independent* if

$$\alpha_1 v_1 + \alpha_2 v_2 + \cdots + \alpha_k v_k = 0 \Longrightarrow \alpha_1 = \alpha_2 = \cdots = 0$$

some equivalent conditions:

- coefficients of $\alpha_1 v_1 + \alpha_2 v_2 + \cdots + \alpha_k v_k$ are uniquely determined, *i.e.*,

  $$\alpha_1 v_1 + \alpha_2 v_2 + \cdots + \alpha_k v_k = \beta_1 v_1 + \beta_2 v_2 + \cdots + \beta_k v_k$$

  implies $\alpha_1 = \beta_1, \alpha_2 = \beta_2, \ldots, \alpha_k = \beta_k$

- no vector v_i can be expressed as a linear combination of the other vectors $v_1, \ldots, v_{i-1}, v_{i+1}, \ldots, v_k$

Basis and dimension

set of vectors $\{v_1, v_2, \ldots, v_k\}$ is a *basis* for a vector space $\mathcal{V}$ if

- $v_1, v_2, \ldots, v_k$ span $\mathcal{V}$, *i.e.*, $\mathcal{V} = \mathrm{span}(v_1, v_2, \ldots, v_k)$
- $\{v_1, v_2, \ldots, v_k\}$ is independent

equivalent: every $v \in \mathcal{V}$ can be uniquely expressed as

$$v = \alpha_1 v_1 + \cdots + \alpha_k v_k$$

fact: for a given vector space $\mathcal{V}$, the number of vectors in any basis is the same

number of vectors in any basis is called the *dimension* of $\mathcal{V}$, denoted $\mathbf{dim}\mathcal{V}$

(we assign $\mathbf{dim}\{0\} = 0$, and $\mathbf{dim}\mathcal{V} = \infty$ if there is no basis)

Nullspace of a matrix

the *nullspace* of $A \in \mathbf{R}^{m \times n}$ is defined as

$$\mathcal{N}(A) = \{ \, x \in \mathbf{R}^n \mid Ax = 0 \, \}$$

- $\mathcal{N}(A)$ is set of vectors mapped to zero by $y = Ax$
- $\mathcal{N}(A)$ is set of vectors orthogonal to all rows of A

$\mathcal{N}(A)$ gives *ambiguity* in x given $y = Ax$:

- if $y = Ax$ and $z \in \mathcal{N}(A)$, then $y = A(x + z)$
- conversely, if $y = Ax$ and $y = A\tilde{x}$, then $\tilde{x} = x + z$ for some $z \in \mathcal{N}(A)$

Zero nullspace

A is called *one-to-one* if 0 is the only element of its nullspace:
$\mathcal{N}(A) = \{0\} \iff$

- x can always be uniquely determined from $y = Ax$ ($i.e.$, the linear transformation $y = Ax$ doesn't 'lose' information)
- mapping from x to Ax is one-to-one: different x's map to different y's
- columns of A are independent (hence, a basis for their span)
- A has a *left inverse*, $i.e.$, there is a matrix $B \in \mathbf{R}^{n \times m}$ s.t. $BA = I$
- $\det(A^TA) \neq 0$

(we'll establish these later)

Interpretations of nullspace

suppose $z \in \mathcal{N}(A)$

$y = Ax$ represents **measurement** of x

- z is undetectable from sensors — get zero sensor readings
- x and $x + z$ are indistinguishable from sensors: $Ax = A(x + z)$

$\mathcal{N}(A)$ characterizes *ambiguity* in x from measurement $y = Ax$

$y = Ax$ represents **output** resulting from input x

- z is an input with no result
- x and $x + z$ have same result

$\mathcal{N}(A)$ characterizes *freedom of input choice* for given result

Range of a matrix

the *range* of $A \in \mathbf{R}^{m \times n}$ is defined as

$$\mathcal{R}(A) = \{Ax \mid x \in \mathbf{R}^n\} \subseteq \mathbf{R}^m$$

$\mathcal{R}(A)$ can be interpreted as

- the set of vectors that can be 'hit' by linear mapping $y = Ax$
- the span of columns of A
- the set of vectors y for which $Ax = y$ has a solution

Onto matrices

A is called *onto* if $\mathcal{R}(A) = \mathbf{R}^m \iff$

- $Ax = y$ can be solved in x for any y
- columns of A span $\mathbf{R}^m$
- A has a *right inverse*, $i.e.$, there is a matrix $B \in \mathbf{R}^{n\times m}$ s.t. $AB = I$
- rows of A are independent
- $\mathcal{N}(A^T) = \{0\}$
- $\det(AA^T) \neq 0$

(some of these are not obvious; we'll establish them later)

Interpretations of range

suppose $v \in \mathcal{R}(A)$, $w \notin \mathcal{R}(A)$

$y = Ax$ represents **measurement** of x

- $y = v$ is a *possible* or *consistent* sensor signal
- $y = w$ is *impossible* or *inconsistent*; sensors have failed or model is wrong

$y = Ax$ represents **output** resulting from input x

- v is a possible result or output
- w cannot be a result or output

$\mathcal{R}(A)$ characterizes the *possible results* or *achievable outputs*

Inverse

$A \in \mathbf{R}^{n \times n}$ is *invertible* or *nonsingular* if $\det A \neq 0$

equivalent conditions:

- columns of A are a basis for $\mathbf{R}^n$
- rows of A are a basis for $\mathbf{R}^n$
- $y = Ax$ has a unique solution x for every $y \in \mathbf{R}^n$
- A has a (left and right) inverse denoted $A^{-1} \in \mathbf{R}^{n \times n}$, with $AA^{-1} = A^{-1}A = I$
- $\mathcal{N}(A) = \{0\}$
- $\mathcal{R}(A) = \mathbf{R}^n$
- $\det A^T A = \det AA^T \neq 0$

Interpretations of inverse

suppose $A \in \mathbf{R}^{n \times n}$ has inverse $B = A^{-1}$

- mapping associated with B undoes mapping associated with A (applied either before or after!)
- $x = By$ is a perfect (pre- or post-) *equalizer* for the *channel* $y = Ax$
- $x = By$ is unique solution of $Ax = y$

Dual basis interpretation

- let a_i be columns of A, and $\tilde{b}_i^T$ be rows of $B = A^{-1}$
- from $y = x_1 a_1 + \cdots + x_n a_n$ and $x_i = \tilde{b}_i^T y$, we get

$$y = \sum_{i=1}^{n} (\tilde{b}_i^T y) a_i$$

  thus, inner product with *rows of inverse matrix* gives the coefficients in the *expansion of a vector in the columns of the matrix*

- $\tilde{b}_1, \ldots, \tilde{b}_n$ and $a_1, \ldots, a_n$ are called *dual bases*

Rank of a matrix

we define the *rank* of $A \in \mathbf{R}^{m \times n}$ as

$$\mathbf{rank}(A) = \mathbf{dim}\, \mathcal{R}(A)$$

(nontrivial) **facts:**

- $\mathbf{rank}(A) = \mathbf{rank}(A^T)$
- $\mathbf{rank}(A)$ is maximum number of independent columns (or rows) of A hence $\mathbf{rank}(A) \leq \mathbf{min}(m, n)$
- $\mathbf{rank}(A) + \mathbf{dim}\, \mathcal{N}(A) = n$

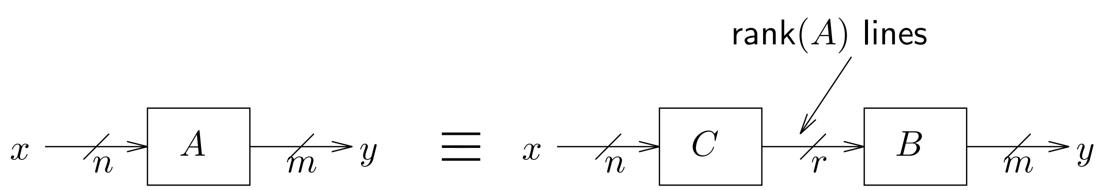
Conservation of dimension

interpretation of $\mathbf{rank}(A) + \mathbf{dim}\,\mathcal{N}(A) = n$:

- $\mathbf{rank}(A)$ is dimension of set 'hit' by the mapping $y = Ax$
- $\mathbf{dim}\,\mathcal{N}(A)$ is dimension of set of x 'crushed' to zero by $y = Ax$
- 'conservation of dimension': each dimension of input is either crushed to zero or ends up in output
- roughly speaking:
  - n is number of degrees of freedom in input x
  - $\mathbf{dim}\,\mathcal{N}(A)$ is number of degrees of freedom lost in the mapping from x to $y = Ax$
  - $\mathbf{rank}(A)$ is number of degrees of freedom in output y

'Coding' interpretation of rank

- rank of product: $\mathbf{rank}(BC) \le \mathbf{min}\{\mathbf{rank}(B),\ \mathbf{rank}(C)\}$
- hence if $A = BC$ with $B \in \mathbf{R}^{m \times r}$, $C \in \mathbf{R}^{r \times n}$, then $\mathbf{rank}(A) \le r$
- conversely: if $\mathbf{rank}(A) = r$ then $A \in \mathbf{R}^{m \times n}$ can be factored as $A = BC$ with $B \in \mathbf{R}^{m \times r}$, $C \in \mathbf{R}^{r \times n}$:



- $\mathbf{rank}(A) = r$ is minimum size of vector needed to faithfully reconstruct y from x

Application: fast matrix-vector multiplication

- need to compute matrix-vector product $y = Ax$, $A \in \mathbf{R}^{m \times n}$
- A has known factorization $A = BC$, $B \in \mathbf{R}^{m \times r}$
- computing $y = Ax$ directly: mn operations
- computing $y = Ax$ as $y = B(Cx)$ (compute $z = Cx$ first, then $y = Bz$): $rn + mr = (m + n)r$ operations
- savings can be considerable if $r \ll \min\{m, n\}$

Full rank matrices

for $A \in \mathbf{R}^{m \times n}$ we always have $\mathbf{rank}(A) \leq \mathbf{min}(m, n)$

we say A is *full rank* if $\mathbf{rank}(A) = \mathbf{min}(m, n)$

- for **square** matrices, full rank means nonsingular
- for **skinny** matrices ($m \geq n$), full rank means columns are independent
- for **fat** matrices ($m \leq n$), full rank means rows are independent

Change of coordinates

'standard' basis vectors in $\mathbf{R}^n$: $(e_1, e_2, \ldots, e_n)$ where

$$e_i = \begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix}$$

(1 in ith component)

obviously we have

$$x = x_1 e_1 + x_2 e_2 + \cdots + x_n e_n$$

x_i are called the coordinates of x (in the standard basis)

if $(t_1, t_2, \ldots, t_n)$ is another basis for $\mathbf{R}^n$, we have

$$x = \tilde{x}_1 t_1 + \tilde{x}_2 t_2 + \cdots + \tilde{x}_n t_n$$

where $\tilde{x}_i$ are the coordinates of x in the basis $(t_1, t_2, \ldots, t_n)$

define $T = \begin{bmatrix} t_1 & t_2 & \cdots & t_n \end{bmatrix}$ so $x = T\tilde{x}$, hence

$$\tilde{x} = T^{-1} x$$

(T is invertible since t_i are a basis)

T^{-1} transforms (standard basis) coordinates of x into t_i-coordinates

inner product ith row of T^{-1} with x extracts t_i-coordinate of x

consider linear transformation $y = Ax$, $A \in \mathbf{R}^{n \times n}$

express y and x in terms of $t_1, t_2 \ldots, t_n$:

$$x = T\tilde{x}, \quad y = T\tilde{y}$$

so

$$\tilde{y} = (T^{-1}AT)\tilde{x}$$

- $A \longrightarrow T^{-1}AT$ is called *similarity transformation*
- similarity transformation by T expresses linear transformation $y = Ax$ in coordinates $t_1, t_2, \ldots, t_n$

(Euclidean) norm

for $x \in \mathbf{R}^n$ we define the (Euclidean) norm as

$$\|x\| = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2} = \sqrt{x^T x}$$

$\|x\|$ measures length of vector (from origin)

important properties:

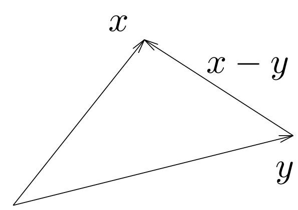
- $\|\alpha x\| = |\alpha|\|x\|$ (homogeneity)
- $\|x + y\| \leq \|x\| + \|y\|$ (triangle inequality)
- $\|x\| \geq 0$ (nonnegativity)
- $\|x\| = 0 \Longleftrightarrow x = 0$ (definiteness)

RMS value and (Euclidean) distance

root-mean-square (RMS) value of vector $x \in \mathbf{R}^n$:

$$\mathbf{rms}(x) = \left(\frac{1}{n} \sum_{i=1}^{n} x_i^2 \right)^{1/2} = \frac{\|x\|}{\sqrt{n}}$$

norm defines distance between vectors: $\mathbf{dist}(x, y) = \|x - y\|$



Inner product

$$\langle x, y \rangle := x_1 y_1 + x_2 y_2 + \cdots + x_n y_n = x^T y$$

important properties:

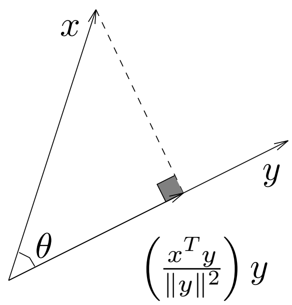
- $\langle \alpha x, y \rangle = \alpha \langle x, y \rangle$
- $\langle x + y, z \rangle = \langle x, z \rangle + \langle y, z \rangle$
- $\langle x, y \rangle = \langle y, x \rangle$
- $\langle x, x \rangle \geq 0$
- $\langle x, x \rangle = 0 \iff x = 0$

$f(y) = \langle x, y \rangle$ is linear function : $\mathbf{R}^n \to \mathbf{R}$, with linear map defined by row vector x^T

Cauchy-Schwartz inequality and angle between vectors

- for any $x, y \in \mathbf{R}^n$, $|x^T y| \leq \|x\|\|y\|$
- (unsigned) angle between vectors in $\mathbf{R}^n$ defined as

$$\theta = \angle(x, y) = \cos^{-1} \frac{x^T y}{\|x\|\|y\|}$$



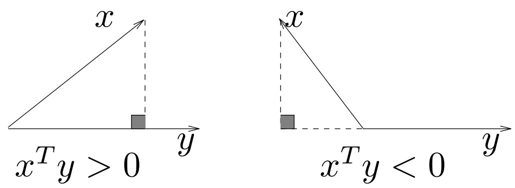
thus $x^T y = \|x\|\|y\| \cos\theta$

special cases:

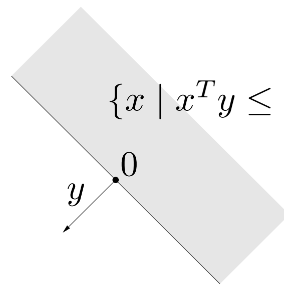
- x and y are *aligned*: $\theta = 0$; $x^T y = \|x\|\|y\|$;
  (if $x \neq 0$) $y = \alpha x$ for some $\alpha \geq 0$
- x and y are *opposed*: $\theta = \pi$; $x^T y = -\|x\|\|y\|$
  (if $x \neq 0$) $y = -\alpha x$ for some $\alpha \geq 0$
- x and y are *orthogonal*: $\theta = \pi/2$ or $-\pi/2$; $x^T y = 0$
  denoted $x \perp y$

interpretation of $x^T y > 0$ and $x^T y < 0$:

- $x^T y > 0$ means $\angle(x, y)$ is acute
- $x^T y < 0$ means $\angle(x, y)$ is obtuse



$\{x \mid x^T y \leq 0\}$ defines a *halfspace* with outward normal vector y, and boundary passing through 0



Lecture 4
Orthonormal sets of vectors and QR factorization

- orthonormal sets of vectors
- Gram-Schmidt procedure, QR factorization
- orthogonal decomposition induced by a matrix

Orthonormal set of vectors

set of vectors $u_1, \ldots, u_k \in \mathbf{R}^n$ is

- *normalized* if $\|u_i\| = 1$, $i = 1, \ldots, k$
  (u_i are called *unit vectors* or *direction vectors*)
- *orthogonal* if $u_i \perp u_j$ for $i \neq j$
- *orthonormal* if both

slang: we say '$u_1, \ldots, u_k$ are orthonormal vectors' but orthonormality (like independence) is a property of a set of vectors, not vectors individually

in terms of $U = [u_1 \ \cdots \ u_k]$, orthonormal means

$$U^T U = I_k$$

- orthonormal vectors are independent
  (multiply $\alpha_1 u_1 + \alpha_2 u_2 + \cdots + \alpha_k u_k = 0$ by u_i^T)

- hence $u_1, \ldots, u_k$ is an orthonormal basis for

$$\operatorname{span}(u_1, \ldots, u_k) = \mathcal{R}(U)$$

- **warning**: if $k < n$ then $UU^T \neq I$ (since its rank is at most k)
  (more on this matrix later . . .)

Geometric properties

suppose columns of $U = [u_1 \ \cdots \ u_k]$ are orthonormal

if $w = Uz$, then $\|w\| = \|z\|$

- multiplication by U does not change norm

- mapping $w = Uz$ is *isometric*: it preserves distances

- simple derivation using matrices:

$$\|w\|^2 = \|Uz\|^2 = (Uz)^T(Uz) = z^T U^T U z = z^T z = \|z\|^2$$

- *inner products* are also preserved: $\langle Uz, U\tilde{z}\rangle = \langle z, \tilde{z}\rangle$

- if $w = Uz$ and $\tilde{w} = U\tilde{z}$ then

$$\langle w, \tilde{w}\rangle = \langle Uz, U\tilde{z}\rangle = (Uz)^T(U\tilde{z}) = z^T U^T U\tilde{z} = \langle z, \tilde{z}\rangle$$

- norms and inner products preserved, so *angles* are preserved: $\angle(Uz, U\tilde{z}) = \angle(z, \tilde{z})$

- thus, multiplication by U preserves inner products, angles, and distances

Orthonormal basis for $\mathbf{R}^n$

- suppose $u_1, \ldots, u_n$ is an orthonormal *basis* for $\mathbf{R}^n$

- then $U = [u_1 \cdots u_n]$ is called **orthogonal**: it is square and satisfies $U^T U = I$

  (you'd think such matrices would be called *orthonormal*, not *orthogonal*)

- it follows that $U^{-1} = U^T$, and hence also $UU^T = I$, *i.e.*,

$$\sum_{i=1}^{n} u_i u_i^T = I$$

Expansion in orthonormal basis

suppose U is orthogonal, so $x = UU^T x$, *i.e.*,

$$x = \sum_{i=1}^{n} (u_i^T x) u_i$$

- $u_i^T x$ is called the *component* of x in the direction u_i
- $a = U^T x$ *resolves* x into the vector of its u_i components
- $x = Ua$ *reconstitutes* x from its u_i components
- $x = Ua = \displaystyle\sum_{i=1}^{n} a_i u_i$ is called the (u_i-) *expansion* of x

the identity $I = UU^T = \sum_{i=1}^n u_i u_i^T$ is sometimes written (in physics) as

$$I = \sum_{i=1}^{n} |u_i\rangle\langle u_i|$$

since

$$x = \sum_{i=1}^{n} |u_i\rangle\langle u_i|x\rangle$$

(but we won't use this notation)

Geometric interpretation

if U is orthogonal, then transformation $w = Uz$

- preserves *norm* of vectors, $i.e.$, $\|Uz\| = \|z\|$
- preserves *angles* between vectors, $i.e.$, $\angle(Uz, U\tilde{z}) = \angle(z, \tilde{z})$

examples:

- rotations (about some axis)
- reflections (through some plane)

Example: rotation by θ in $\mathbf{R}^2$ is given by

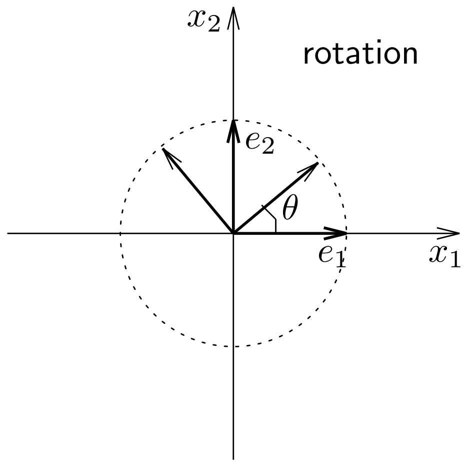
$$y = U_\theta x, \quad U_\theta = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

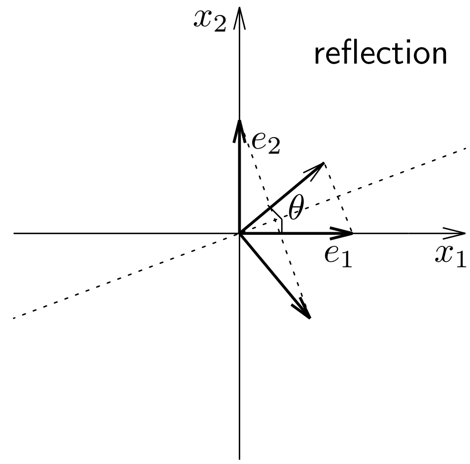
since $e_1 \to (\cos\theta, \sin\theta)$, $e_2 \to (-\sin\theta, \cos\theta)$

reflection across line $x_2 = x_1 \tan(\theta/2)$ is given by

$$y = R_\theta x, \quad R_\theta = \begin{bmatrix} \cos\theta & \sin\theta \\ \sin\theta & -\cos\theta \end{bmatrix}$$

since $e_1 \to (\cos\theta, \sin\theta)$, $e_2 \to (\sin\theta, -\cos\theta)$





can check that U_θ and R_θ are orthogonal

Gram-Schmidt procedure

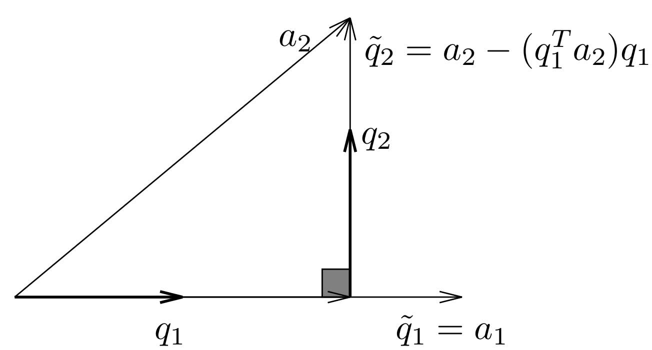
- given independent vectors $a_1, \ldots, a_k \in \mathbf{R}^n$, G-S procedure finds orthonormal vectors $q_1, \ldots, q_k$ s.t.

$$\mathrm{span}(a_1, \ldots, a_r) = \mathrm{span}(q_1, \ldots, q_r) \quad \text{for} \quad r \leq k$$

- thus, $q_1, \ldots, q_r$ is an orthonormal basis for $\mathrm{span}(a_1, \ldots, a_r)$
- rough idea of method: first *orthogonalize* each vector w.r.t. previous ones; then *normalize* result to have norm one

Gram-Schmidt procedure

- step 1a. $\tilde{q}_1 := a_1$
- step 1b. $q_1 := \tilde{q}_1/\|\tilde{q}_1\|$ (normalize)
- step 2a. $\tilde{q}_2 := a_2 - (q_1^T a_2)q_1$ (remove q_1 component from a_2)
- step 2b. $q_2 := \tilde{q}_2/\|\tilde{q}_2\|$ (normalize)
- step 3a. $\tilde{q}_3 := a_3 - (q_1^T a_3)q_1 - (q_2^T a_3)q_2$ (remove q_1, q_2 components)
- step 3b. $q_3 := \tilde{q}_3/\|\tilde{q}_3\|$ (normalize)
- etc.



for $i = 1, 2, \ldots, k$ we have

$$\begin{aligned} a_i &= (q_1^T a_i)q_1 + (q_2^T a_i)q_2 + \cdots + (q_{i-1}^T a_i)q_{i-1} + \|\tilde{q}_i\| q_i \\ &= r_{1i}q_1 + r_{2i}q_2 + \cdots + r_{ii}q_i \end{aligned}$$

(note that the r_{ij}'s come right out of the G-S procedure, and $r_{ii} \neq 0$)

QR decomposition

written in matrix form: $A = QR$, where $A \in \mathbf{R}^{n\times k}$, $Q \in \mathbf{R}^{n\times k}$, $R \in \mathbf{R}^{k\times k}$:

$$\underbrace{\begin{bmatrix} a_1 & a_2 & \cdots & a_k \end{bmatrix}}_{A} = \underbrace{\begin{bmatrix} q_1 & q_2 & \cdots & q_k \end{bmatrix}}_{Q} \underbrace{\begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1k} \\ 0 & r_{22} & \cdots & r_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & r_{kk} \end{bmatrix}}_{R}$$

- $Q^TQ = I_k$, and R is upper triangular & invertible
- called QR **decomposition** (or factorization) of A
- usually computed using a variation on Gram-Schmidt procedure which is less sensitive to numerical (rounding) errors
- columns of Q are orthonormal basis for $\mathcal{R}(A)$

General Gram-Schmidt procedure

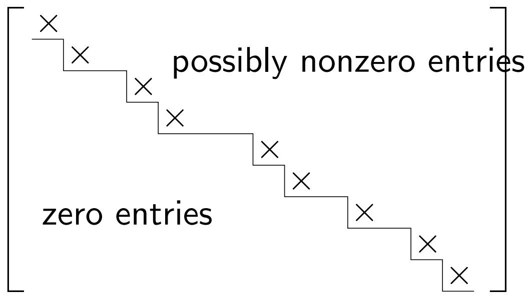
- in basic G-S we assume $a_1, \ldots, a_k \in \mathbf{R}^n$ are independent
- if $a_1, \ldots, a_k$ are dependent, we find $\tilde{q}_j = 0$ for some j, which means a_j is linearly dependent on $a_1, \ldots, a_{j-1}$
- modified algorithm: when we encounter $\tilde{q}_j = 0$, skip to next vector a_{j+1} and continue:

$r = 0$;
for $i = 1, \ldots, k$
$\{$
  $\tilde{a} = a_i - \sum_{j=1}^{r} q_j q_j^T a_i$;
  if $\tilde{a} \neq 0$ $\{$ $r = r + 1$; $q_r = \tilde{a}/\|\tilde{a}\|$; $\}$
$\}$

on exit,

- $q_1, \ldots, q_r$ is an orthonormal basis for $\mathcal{R}(A)$ (hence $r = \mathbf{Rank}(A)$)
- each a_i is linear combination of previously generated q_j's

in matrix notation we have $A = QR$ with $Q^TQ = I_r$ and $R \in \mathbf{R}^{r \times k}$ in *upper staircase form*:



'corner' entries (shown as $\times$) are nonzero

can permute columns with $\times$ to front of matrix:

$$A = Q[\tilde{R}\ S]P$$

where:

- $Q^TQ = I_r$
- $\tilde{R} \in \mathbf{R}^{r \times r}$ is upper triangular and invertible
- $P \in \mathbf{R}^{k \times k}$ is a permutation matrix
  (which moves forward the columns of a which generated a new q)

Applications

- directly yields orthonormal basis for $\mathcal{R}(A)$
- yields factorization $A = BC$ with $B \in \mathbf{R}^{n\times r}$, $C \in \mathbf{R}^{r\times k}$, $r = \mathbf{Rank}(A)$
- to check if $b \in \mathrm{span}(a_1, \ldots, a_k)$: apply Gram-Schmidt to $[a_1 \ \cdots \ a_k \ b]$
- staircase pattern in R shows which columns of A are dependent on previous ones

works incrementally: one G-S procedure yields QR factorizations of $[a_1 \ \cdots \ a_p]$ for $p = 1, \ldots, k$:

$$[a_1 \ \cdots \ a_p] = [q_1 \ \cdots \ q_s] R_p$$

where $s = \mathbf{Rank}([a_1 \ \cdots \ a_p])$ and R_p is leading $s \times p$ submatrix of R

'Full' QR factorization

with $A = Q_1 R_1$ the QR factorization as above, write

$$A = \left[\begin{array}{cc} Q_1 & Q_2 \end{array} \right] \left[\begin{array}{c} R_1 \\ 0 \end{array} \right]$$

where $[Q_1 \ Q_2]$ is orthogonal, $i.e.$, columns of $Q_2 \in \mathbf{R}^{n\times(n-r)}$ are orthonormal, orthogonal to Q_1

to find Q_2:

- find any matrix $\tilde{A}$ s.t. $[A \ \tilde{A}]$ is full rank ($e.g.$, $\tilde{A} = I$)
- apply general Gram-Schmidt to $[A \ \tilde{A}]$
- Q_1 are orthonormal vectors obtained from columns of A
- Q_2 are orthonormal vectors obtained from extra columns ($\tilde{A}$)

$i.e.$, any set of orthonormal vectors can be *extended* to an orthonormal basis for $\mathbf{R}^n$

$\mathcal{R}(Q_1)$ and $\mathcal{R}(Q_2)$ are called *complementary subspaces* since

- they are orthogonal ($i.e.$, every vector in the first subspace is orthogonal to every vector in the second subspace)
- their sum is $\mathbf{R}^n$ ($i.e.$, every vector in $\mathbf{R}^n$ can be expressed as a sum of two vectors, one from each subspace)

this is written

- $\mathcal{R}(Q_1) \overset{\perp}{+} \mathcal{R}(Q_2) = \mathbf{R}^n$
- $\mathcal{R}(Q_2) = \mathcal{R}(Q_1)^\perp$ (and $\mathcal{R}(Q_1) = \mathcal{R}(Q_2)^\perp$)
  (each subspace is the *orthogonal complement* of the other)

we know $\mathcal{R}(Q_1) = \mathcal{R}(A)$; but what is its orthogonal complement $\mathcal{R}(Q_2)$?

Orthogonal decomposition induced by A

from $A^T = \begin{bmatrix} R_1^T & 0 \end{bmatrix} \begin{bmatrix} Q_1^T \\ Q_2^T \end{bmatrix}$ we see that

$$A^T z = 0 \iff Q_1^T z = 0 \iff z \in \mathcal{R}(Q_2)$$

so $\mathcal{R}(Q_2) = \mathcal{N}(A^T)$
(in fact the columns of Q_2 are an orthonormal basis for $\mathcal{N}(A^T)$)

we conclude: $\mathcal{R}(A)$ and $\mathcal{N}(A^T)$ are *complementary subspaces*:

- $\mathcal{R}(A) \overset{\perp}{+} \mathcal{N}(A^T) = \mathbf{R}^n$ (recall $A \in \mathbf{R}^{n \times k}$)
- $\mathcal{R}(A)^\perp = \mathcal{N}(A^T)$ (and $\mathcal{N}(A^T)^\perp = \mathcal{R}(A)$)
- called *orthogonal decomposition (of $\mathbf{R}^n$) induced by $A \in \mathbf{R}^{n \times k}$*

- every $y \in \mathbf{R}^n$ can be written uniquely as $y = z + w$, with $z \in \mathcal{R}(A)$, $w \in \mathcal{N}(A^T)$ (we'll soon see what the vector z is . . .)

- can now prove most of the assertions from the linear algebra review lecture

- switching $A \in \mathbf{R}^{n \times k}$ to $A^T \in \mathbf{R}^{k \times n}$ gives decomposition of $\mathbf{R}^k$:

$$\mathcal{N}(A) \overset{\perp}{+} \mathcal{R}(A^T) = \mathbf{R}^k$$

Lecture 5
Least-squares

- least-squares (approximate) solution of overdetermined equations
- projection and orthogonality principle
- least-squares estimation
- BLUE property

Overdetermined linear equations

consider $y = Ax$ where $A \in \mathbf{R}^{m \times n}$ is (strictly) skinny, $i.e.$, $m > n$

- called *overdetermined* set of linear equations
  (more equations than unknowns)
- for most y, cannot solve for x

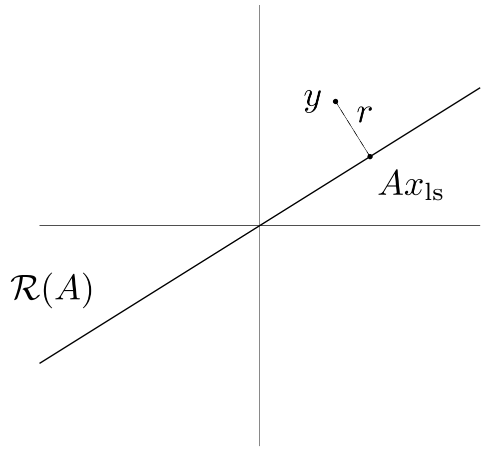
one approach to *approximately* solve $y = Ax$:

- define *residual* or error $r = Ax - y$
- find $x = x_{\mathrm{ls}}$ that minimizes $\|r\|$

x_{ls} called *least-squares* (approximate) solution of $y = Ax$

Geometric interpretation

Ax_{ls} is point in $\mathcal{R}(A)$ closest to y (Ax_{ls} is *projection* of y onto $\mathcal{R}(A)$)



Least-squares (approximate) solution

- assume A is full rank, skinny
- to find x_{ls}, we'll minimize norm of residual squared,

$$\|r\|^2 = x^T A^T A x - 2y^T A x + y^T y$$

- set gradient w.r.t. x to zero:

$$\nabla_x \|r\|^2 = 2A^T A x - 2A^T y = 0$$

- yields the *normal equations*: $A^T A x = A^T y$
- assumptions imply $A^T A$ invertible, so we have

$$x_{\mathrm{ls}} = (A^T A)^{-1} A^T y$$

. . . a very famous formula

- $x_{\rm ls}$ is linear function of y
- $x_{\rm ls} = A^{-1}y$ if A is square
- $x_{\rm ls}$ solves $y = Ax_{\rm ls}$ if $y \in \mathcal{R}(A)$
- $A^\dagger = (A^TA)^{-1}A^T$ is called the *pseudo-inverse* of A
- $A^\dagger$ is a *left inverse* of (full rank, skinny) A:

$$A^\dagger A = (A^TA)^{-1}A^T\,A = I$$

Projection on $\mathcal{R}(A)$

$Ax_{\rm ls}$ is (by definition) the point in $\mathcal{R}(A)$ that is closest to y, $i.e.$, it is the *projection* of y onto $\mathcal{R}(A)$

$$Ax_{\rm ls} = \mathcal{P}_{\mathcal{R}(A)}(y)$$

- the projection function $\mathcal{P}_{\mathcal{R}(A)}$ is linear, and given by

$$\mathcal{P}_{\mathcal{R}(A)}(y) = Ax_{\rm ls} = A(A^TA)^{-1}A^Ty$$

- $A(A^TA)^{-1}A^T$ is called the *projection matrix* (associated with $\mathcal{R}(A)$)

Orthogonality principle

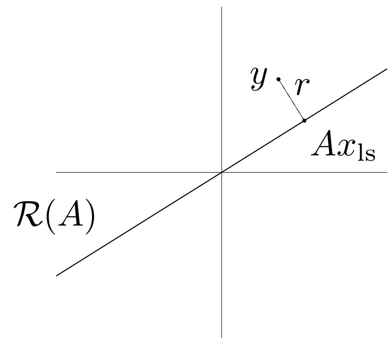
optimal residual

$$r = Ax_{\rm ls} - y = (A(A^TA)^{-1}A^T - I)y$$

is orthogonal to $\mathcal{R}(A)$:

$$\langle r, Az\rangle = y^T(A(A^TA)^{-1}A^T - I)^TAz = 0$$

for all $z \in \mathbf{R}^n$



Completion of squares

since $r = Ax_{\rm ls} - y \perp A(x - x_{\rm ls})$ for any x, we have

$$\begin{aligned}\|Ax - y\|^2 &= \|(Ax_{\rm ls} - y) + A(x - x_{\rm ls})\|^2 \\ &= \|Ax_{\rm ls} - y\|^2 + \|A(x - x_{\rm ls})\|^2\end{aligned}$$

this shows that for $x \neq x_{\rm ls}$, $\|Ax - y\| > \|Ax_{\rm ls} - y\|$

Least-squares via QR factorization

- $A \in \mathbf{R}^{m\times n}$ skinny, full rank
- factor as $A = QR$ with $Q^TQ = I_n$, $R \in \mathbf{R}^{n\times n}$ upper triangular, invertible
- pseudo-inverse is

$$(A^TA)^{-1}A^T = (R^TQ^TQR)^{-1}R^TQ^T = R^{-1}Q^T$$

  so $x_{\rm ls} = R^{-1}Q^Ty$

- projection on $\mathcal{R}(A)$ given by matrix

$$A(A^TA)^{-1}A^T = AR^{-1}Q^T = QQ^T$$

Least-squares via full QR factorization

- full QR factorization:

$$A = [Q_1\ Q_2] \begin{bmatrix} R_1 \\ 0 \end{bmatrix}$$

  with $[Q_1\ Q_2] \in \mathbf{R}^{m\times m}$ orthogonal, $R_1 \in \mathbf{R}^{n\times n}$ upper triangular, invertible

- multiplication by orthogonal matrix doesn't change norm, so

$$\begin{aligned} \|Ax - y\|^2 &= \left\| [Q_1\ Q_2] \begin{bmatrix} R_1 \\ 0 \end{bmatrix} x - y \right\|^2 \\ &= \left\| [Q_1\ Q_2]^T [Q_1\ Q_2] \begin{bmatrix} R_1 \\ 0 \end{bmatrix} x - [Q_1\ Q_2]^T y \right\|^2 \end{aligned}$$

$$
\begin{aligned}
&= \left\| \begin{bmatrix} R_1 x - Q_1^T y \\ -Q_2^T y \end{bmatrix} \right\|^2 \\
&= \|R_1 x - Q_1^T y\|^2 + \|Q_2^T y\|^2
\end{aligned}
$$

- this is evidently minimized by choice $x_{\mathrm{ls}} = R_1^{-1} Q_1^T y$ (which makes first term zero)

- residual with optimal x is

$$
A x_{\mathrm{ls}} - y = -Q_2 Q_2^T y
$$

- $Q_1 Q_1^T$ gives projection onto $\mathcal{R}(A)$

- $Q_2 Q_2^T$ gives projection onto $\mathcal{R}(A)^\perp$

Least-squares estimation

many applications in inversion, estimation, and reconstruction problems have form

$$
y = Ax + v
$$

- x is what we want to estimate or reconstruct

- y is our sensor measurement(s)

- v is an unknown *noise* or *measurement error* (assumed small)

- ith row of A characterizes ith sensor least-squares estimation: choose as estimate $\hat{x}$ that minimizes

$$\|A\hat{x} - y\|$$

$i.e.$, deviation between

- what we actually observed (y), and
- what we would observe if $x = \hat{x}$, and there were no noise ($v = 0$)

least-squares estimate is just $\hat{x} = (A^T A)^{-1} A^T y$

BLUE property

linear measurement with noise:

$$y = Ax + v$$

with A full rank, skinny

consider a *linear estimator* of form $\hat{x} = By$

- called *unbiased* if $\hat{x} = x$ whenever $v = 0$
  ($i.e.$, no estimation error when there is no noise)

  same as $BA = I$, $i.e.$, B is left inverse of A

- estimation error of unbiased linear estimator is

$$x - \hat{x} = x - B(Ax + v) = -Bv$$

obviously, then, we'd like B 'small' (and $BA = I$)

- **fact:** $A^\dagger = (A^TA)^{-1}A^T$ is the *smallest* left inverse of A, in the following sense:

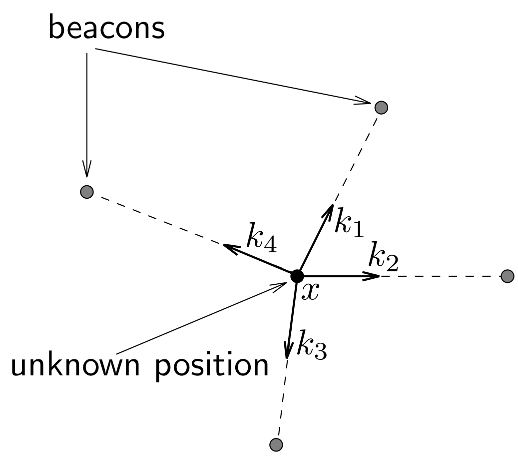
  for any B with $BA = I$, we have

$$\sum_{i,j} B_{ij}^2 \geq \sum_{i,j} A_{ij}^{\dagger 2}$$

  $i.e.$, least-squares provides the *best linear unbiased estimator* (BLUE)

Navigation from range measurements

navigation using range measurements from *distant* beacons



beacons far from unknown position $x \in \mathbf{R}^2$, so linearization around $x = 0$ (say) nearly exact ranges $y \in \mathbf{R}^4$ measured, with measurement noise v:

$$y = -\begin{bmatrix} k_1^T \\ k_2^T \\ k_3^T \\ k_4^T \end{bmatrix} x + v$$

where k_i is unit vector from 0 to beacon i

measurement errors are independent, Gaussian, with standard deviation 2 (details not important)

problem: estimate $x \in \mathbf{R}^2$, given $y \in \mathbf{R}^4$

(roughly speaking, a $2\!:\!1$ measurement redundancy ratio)

actual position is $x = (5.59, 10.58)$;
measurement is $y = (-11.95, -2.84, -9.81, 2.81)$

Just enough measurements method

y_1 and y_2 suffice to find x (when $v = 0$)

compute estimate $\hat{x}$ by inverting top (2×2) half of A:

$$\hat{x} = B_{\mathrm{je}} y = \begin{bmatrix} 0 & -1.0 & 0 & 0 \\ -1.12 & 0.5 & 0 & 0 \end{bmatrix} y = \begin{bmatrix} 2.84 \\ 11.9 \end{bmatrix}$$

(norm of error: 3.07)

Least-squares method

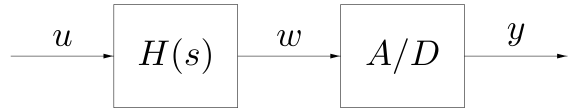
compute estimate $\hat{x}$ by least-squares:

$$\hat{x} = A^\dagger y = \begin{bmatrix} -0.23 & -0.48 & 0.04 & 0.44 \\ -0.47 & -0.02 & -0.51 & -0.18 \end{bmatrix} y = \begin{bmatrix} 4.95 \\ 10.26 \end{bmatrix}$$

(norm of error: 0.72)

- $B_{\rm je}$ and $A^\dagger$ are both left inverses of A
- larger entries in B lead to larger estimation error

Example from overview lecture



- signal u is piecewise constant, period $1\,\mathrm{sec}$, $0 \le t \le 10$:

$$u(t) = x_j, \quad j-1 \le t < j, \quad j = 1, \ldots, 10$$

- filtered by system with impulse response $h(t)$:

$$w(t) = \int_0^t h(t-\tau) u(\tau)\, d\tau$$

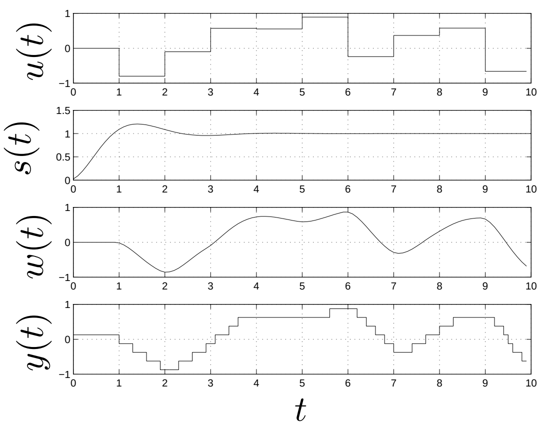
- sample at 10Hz: $\tilde{y}_i = w(0.1i)$, $i = 1, \ldots, 100$

- 3-bit quantization: $y_i = Q(\tilde{y}_i)$, $i = 1, \ldots, 100$, where Q is 3-bit quantizer characteristic

$$Q(a) = (1/4)\left(\mathbf{round}(4a + 1/2) - 1/2\right)$$

- **problem:** estimate $x \in \mathbf{R}^{10}$ given $y \in \mathbf{R}^{100}$

example:

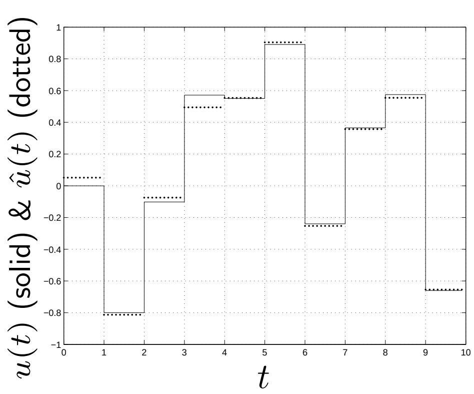


we have $y = Ax + v$, where

- $A \in \mathbf{R}^{100\times 10}$ is given by $A_{ij} = \displaystyle\int_{j-1}^{j} h(0.1i - \tau)\, d\tau$

- $v \in \mathbf{R}^{100}$ is *quantization error:* $v_i = Q(\tilde{y}_i) - \tilde{y}_i$ (so $|v_i| \leq 0.125$)

least-squares estimate: $x_{\rm ls} = (A^TA)^{-1}A^Ty$

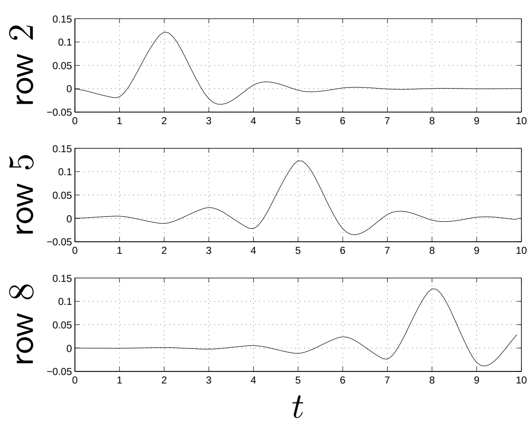


RMS error is $\dfrac{\|x - x_{\rm ls}\|}{\sqrt{10}} = 0.03$

better than if we had no filtering! (RMS error 0.07)

more on this later . . .

some rows of $B_{\rm ls} = (A^T A)^{-1} A^T$:



- rows show how sampled measurements of y are used to form estimate of x_i for $i = 2, 5, 8$
- to estimate x_5, which is the original input signal for $4 \le t < 5$, we mostly use $y(t)$ for $3 \le t \le 7$

Lecture 6
Least-squares applications

- least-squares data fitting
- growing sets of regressors
- system identification
- growing sets of measurements and recursive least-squares

Least-squares data fitting

we are given:

- functions $f_1, \ldots, f_n : S \to \mathbf{R}$, called *regressors* or *basis functions*
- *data* or *measurements* (s_i, g_i), $i = 1, \ldots, m$, where $s_i \in S$ and (usually) $m \gg n$

problem: find coefficients $x_1, \ldots, x_n \in \mathbf{R}$ so that

$$x_1 f_1(s_i) + \cdots + x_n f_n(s_i) \approx g_i, \quad i = 1, \ldots, m$$

$i.e.$, find linear combination of functions that fits data

least-squares fit: choose x to minimize total square fitting error:

$$\sum_{i=1}^{m} \left(x_1 f_1(s_i) + \cdots + x_n f_n(s_i) - g_i\right)^2$$

- using matrix notation, total square fitting error is $\|Ax - g\|^2$, where $A_{ij} = f_j(s_i)$
- hence, least-squares fit is given by

$$x = (A^T A)^{-1} A^T g$$

  (assuming A is skinny, full rank)

- corresponding function is

$$f_{\text{lsfit}}(s) = x_1 f_1(s) + \cdots + x_n f_n(s)$$

- applications:
  - interpolation, extrapolation, smoothing of data
  - developing simple, approximate model of data

Least-squares polynomial fitting

problem: fit polynomial of degree $< n$,

$$p(t) = a_0 + a_1 t + \cdots + a_{n-1} t^{n-1},$$

to data (t_i, y_i), $i = 1, \ldots, m$

- basis functions are $f_j(t) = t^{j-1}$, $j = 1, \ldots, n$
- matrix A has form $A_{ij} = t_i^{j-1}$

$$A = \begin{bmatrix} 1 & t_1 & t_1^2 & \cdots & t_1^{n-1} \\ 1 & t_2 & t_2^2 & \cdots & t_2^{n-1} \\ & \vdots & & & \vdots \\ 1 & t_m & t_m^2 & \cdots & t_m^{n-1} \end{bmatrix}$$

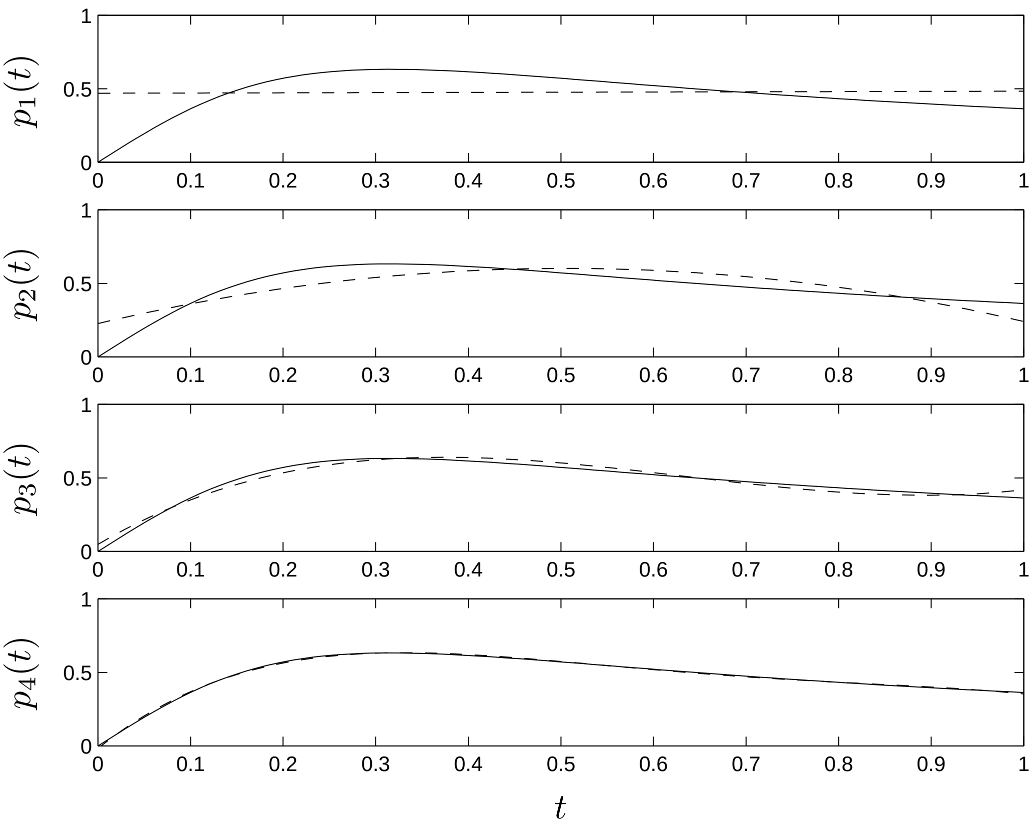
  (called a *Vandermonde matrix*)

assuming $t_k \neq t_l$ for $k \neq l$ and $m \geq n$, A is full rank:

- suppose $Aa = 0$
- corresponding polynomial $p(t) = a_0 + \cdots + a_{n-1}t^{n-1}$ vanishes at m points $t_1, \ldots, t_m$
- by fundamental theorem of algebra p can have no more than $n-1$ zeros, so p is identically zero, and $a = 0$
- columns of A are independent, $i.e.$, A full rank

Example

- fit $g(t) = 4t/(1 + 10t^2)$ with polynomial
- $m = 100$ points between $t = 0$ & $t = 1$
- least-squares fit for degrees 1, 2, 3, 4 have RMS errors $.135$, $.076$, $.025$, $.005$, respectively



Growing sets of regressors

consider *family* of least-squares problems

$$\text{minimize} \quad \|\sum_{i=1}^{p} x_i a_i - y\|$$

for $p = 1, \ldots, n$

($a_1, \ldots, a_p$ are called *regressors*)

- approximate y by linear combination of $a_1, \ldots, a_p$
- project y onto $\text{span}\{a_1, \ldots, a_p\}$
- regress y on $a_1, \ldots, a_p$
- as p increases, get better fit, so optimal residual decreases

solution for each $p \leq n$ is given by

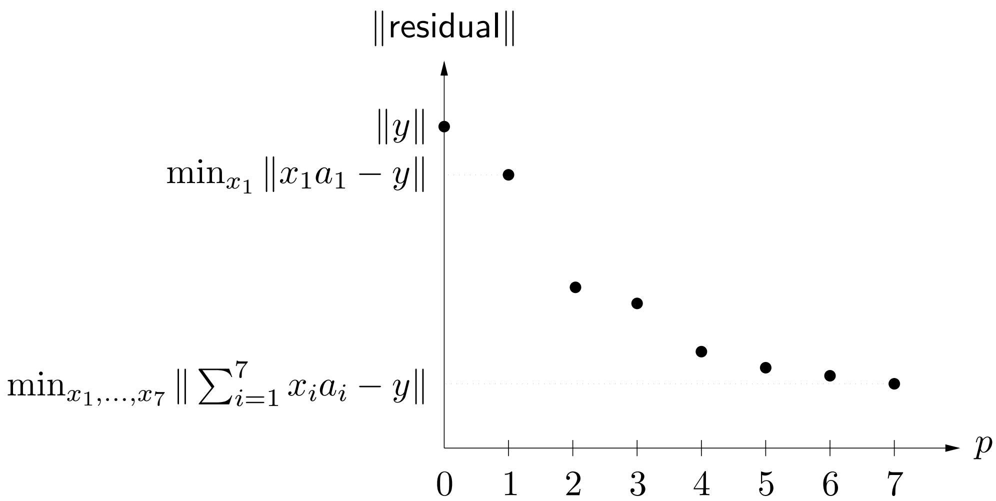
$$x_{\mathrm{ls}}^{(p)} = (A_p^T A_p)^{-1} A_p^T y = R_p^{-1} Q_p^T y$$

where

- $A_p = [a_1 \cdots a_p] \in \mathbf{R}^{m \times p}$ is the first p columns of A
- $A_p = Q_p R_p$ is the QR factorization of A_p
- $R_p \in \mathbf{R}^{p \times p}$ is the leading $p \times p$ submatrix of R
- $Q_p = [q_1 \cdots q_p]$ is the first p columns of Q

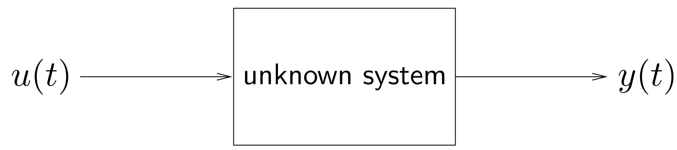
Norm of optimal residual versus p

plot of optimal residual versus p shows how well y can be matched by linear combination of $a_1, \ldots, a_p$, as function of p



Least-squares system identification

we measure input $u(t)$ and output $y(t)$ for $t = 0, \ldots, N$ of unknown system



system identification problem: find reasonable model for system based on measured I/O data u, y

example with scalar u, y (vector u, y readily handled): fit I/O data with moving-average (MA) model with n delays

$$\hat{y}(t) = h_0 u(t) + h_1 u(t-1) + \cdots + h_n u(t-n)$$

where $h_0, \ldots, h_n \in \mathbf{R}$

we can write model or predicted output as

$$\begin{bmatrix} \hat{y}(n) \\ \hat{y}(n+1) \\ \vdots \\ \hat{y}(N) \end{bmatrix} = \begin{bmatrix} u(n) & u(n-1) & \cdots & u(0) \\ u(n+1) & u(n) & \cdots & u(1) \\ \vdots & \vdots & & \vdots \\ u(N) & u(N-1) & \cdots & u(N-n) \end{bmatrix} \begin{bmatrix} h_0 \\ h_1 \\ \vdots \\ h_n \end{bmatrix}$$

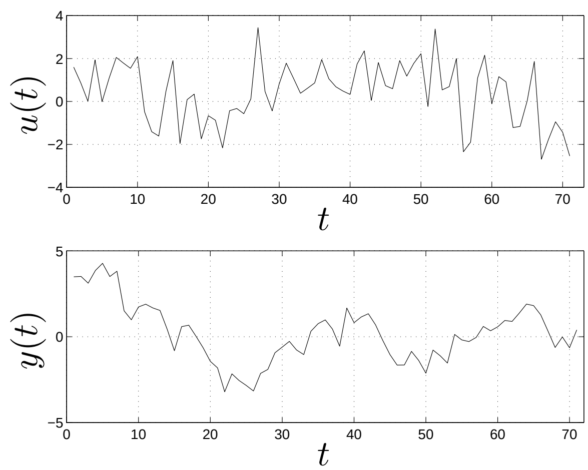
model prediction error is

$$e = (y(n) - \hat{y}(n), \ldots, y(N) - \hat{y}(N))$$

least-squares identification: choose model (*i.e.*, h) that minimizes norm of model prediction error $\|e\|$

. . . a least-squares problem (with variables h)

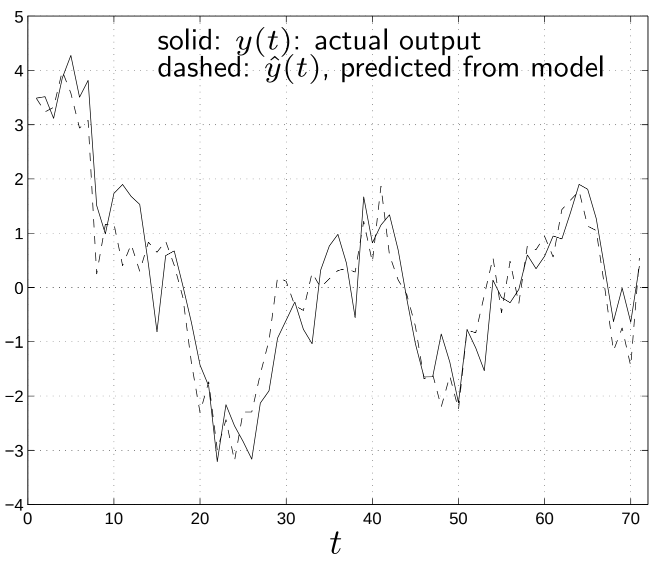
Example



for $n = 7$ we obtain MA model with

$$(h_0, \ldots, h_7) = (.024,\ .282,\ .418,\ .354,\ .243,\ .487,\ .208,\ .441)$$

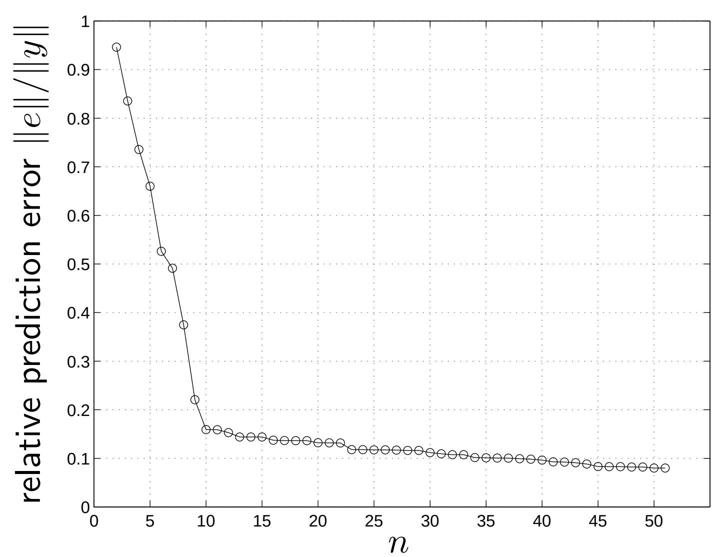
with relative prediction error $\|e\|/\|y\| = 0.37$



Model order selection

question: how large should n be?

- obviously the larger n, the smaller the prediction error *on the data used to form the model*
- suggests using largest possible model order for smallest prediction error

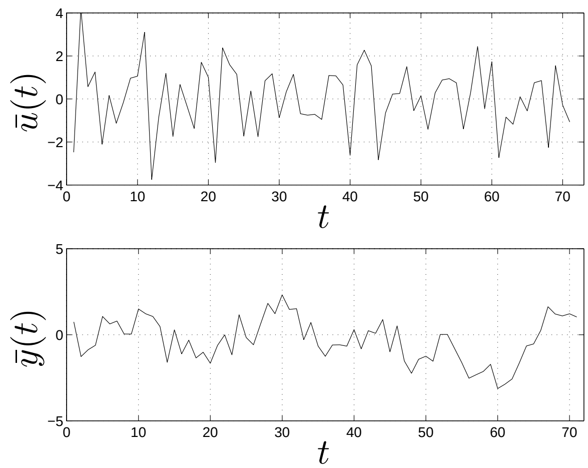


difficulty: for n too large the *predictive ability* of the model on *other I/O data* (from the same system) becomes worse

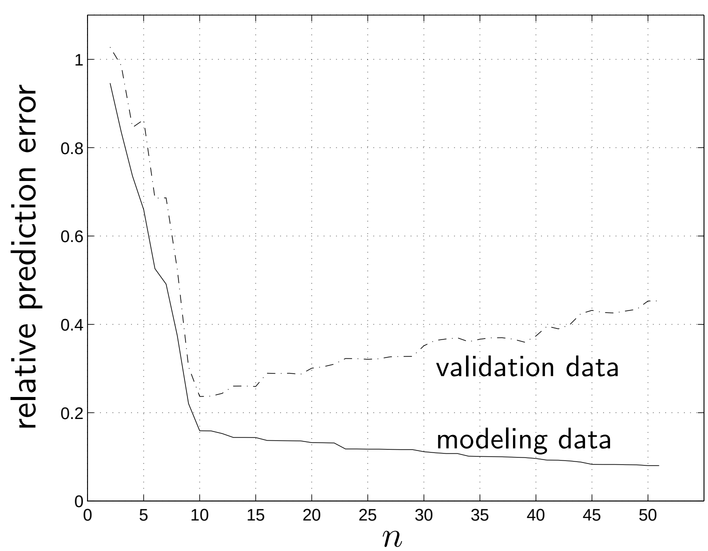
Cross-validation

evaluate model predictive performance on another I/O data set *not used to develop model*

model validation data set:

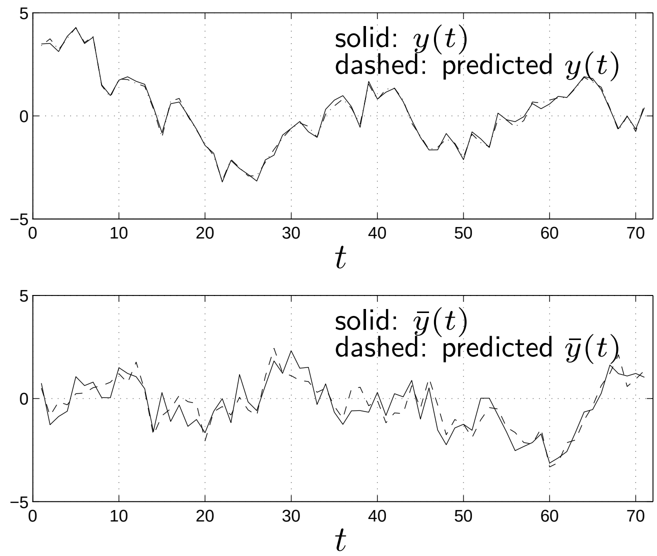


now check prediction error of models (developed using *modeling data*) on *validation data:*



plot suggests $n = 10$ is a good choice

for $n = 50$ the actual and predicted outputs on system identification and model validation data are:



loss of predictive ability when n too large is called *model overfit* or *overmodeling*

Growing sets of measurements

least-squares problem in 'row' form:

$$\text{minimize} \quad \|Ax - y\|^2 = \sum_{i=1}^{m} (a_i^T x - y_i)^2$$

where a_i^T are the rows of A ($a_i \in \mathbf{R}^n$)

- $x \in \mathbf{R}^n$ is some vector to be estimated
- each pair a_i, y_i corresponds to one measurement
- solution is
$$x_{\rm ls} = \left(\sum_{i=1}^{m} a_i a_i^T \right)^{-1} \sum_{i=1}^{m} y_i a_i$$
- suppose that a_i and y_i become available sequentially, $i.e.$, m increases with time

Recursive least-squares

we can compute $x_{\rm ls}(m) = \left(\sum_{i=1}^m a_i a_i^T\right)^{-1} \sum_{i=1}^m y_i a_i$ recursively

- initialize $P(0) = 0 \in \mathbf{R}^{n\times n}$, $q(0) = 0 \in \mathbf{R}^n$
- for $m = 0, 1, \ldots,$

$$P(m+1) = P(m) + a_{m+1}a_{m+1}^T \qquad q(m+1) = q(m) + y_{m+1}a_{m+1}$$

- if $P(m)$ is invertible, we have $x_{\rm ls}(m) = P(m)^{-1}q(m)$
- $P(m)$ is invertible $\iff a_1, \ldots, a_m$ span $\mathbf{R}^n$
(so, once $P(m)$ becomes invertible, it stays invertible)

Fast update for recursive least-squares

we can calculate

$$P(m+1)^{-1} = \left(P(m) + a_{m+1}a_{m+1}^T\right)^{-1}$$

efficiently from $P(m)^{-1}$ using the *rank one update formula*

$$\left(P + aa^T\right)^{-1} = P^{-1} - \frac{1}{1 + a^T P^{-1} a}(P^{-1}a)(P^{-1}a)^T$$

valid when $P = P^T$, and P and $P + aa^T$ are both invertible

- gives an $O(n^2)$ method for computing $P(m+1)^{-1}$ from $P(m)^{-1}$
- standard methods for computing $P(m+1)^{-1}$ from $P(m+1)$ are $O(n^3)$

Verification of rank one update formula

$$
\begin{aligned}
&(P+aa^T)\left(P^{-1}-\frac{1}{1+a^TP^{-1}a}(P^{-1}a)(P^{-1}a)^T\right)\\
&\quad= I+aa^TP^{-1}-\frac{1}{1+a^TP^{-1}a}P(P^{-1}a)(P^{-1}a)^T\\
&\qquad -\frac{1}{1+a^TP^{-1}a}aa^T(P^{-1}a)(P^{-1}a)^T\\
&\quad= I+aa^TP^{-1}-\frac{1}{1+a^TP^{-1}a}aa^TP^{-1}-\frac{a^TP^{-1}a}{1+a^TP^{-1}a}aa^TP^{-1}\\
&\quad= I
\end{aligned}
$$

Lecture 7
Regularized least-squares and Gauss-Newton method

- multi-objective least-squares
- regularized least-squares
- nonlinear least-squares
- Gauss-Newton method

Multi-objective least-squares

in many problems we have two (or more) objectives

- we want $J_1 = \|Ax - y\|^2$ small
- and also $J_2 = \|Fx - g\|^2$ small

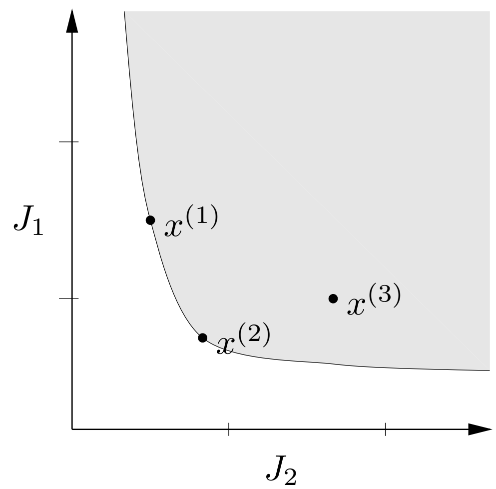
($x \in \mathbf{R}^n$ is the variable)

- usually the objectives are *competing*
- we can make one smaller, at the expense of making the other larger

common example: $F = I$, $g = 0$; we want $\|Ax - y\|$ small, with small x

Plot of achievable objective pairs

plot (J_2, J_1) for every x:



note that $x \in \mathbf{R}^n$, but this plot is in $\mathbf{R}^2$; point labeled $x^{(1)}$ is really $\left(J_2(x^{(1)}), J_1(x^{(1)})\right)$

- shaded area shows (J_2, J_1) achieved by some $x \in \mathbf{R}^n$
- clear area shows (J_2, J_1) not achieved by any $x \in \mathbf{R}^n$
- boundary of region is called *optimal trade-off curve*
- corresponding x are called *Pareto optimal* (for the two objectives $\|Ax - y\|^2$, $\|Fx - g\|^2$)

three example choices of x: $x^{(1)}$, $x^{(2)}$, $x^{(3)}$

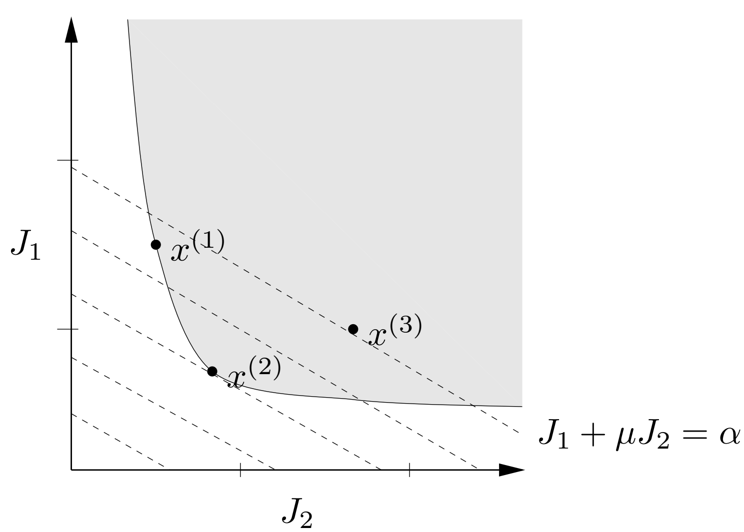
- $x^{(3)}$ is worse than $x^{(2)}$ on both counts (J_2 and J_1)
- $x^{(1)}$ is better than $x^{(2)}$ in J_2, but worse in J_1

Weighted-sum objective

- to find Pareto optimal points, *i.e.*, x's on optimal trade-off curve, we minimize *weighted-sum objective*

$$J_1 + \mu J_2 = \|Ax - y\|^2 + \mu \|Fx - g\|^2$$

- parameter $\mu \geq 0$ gives relative weight between J_1 and J_2
- points where weighted sum is constant, $J_1 + \mu J_2 = \alpha$, correspond to line with slope $-\mu$ on (J_2, J_1) plot



- $x^{(2)}$ minimizes weighted-sum objective for μ shown
- by varying μ from 0 to $+\infty$, can sweep out entire *optimal tradeoff curve*

Minimizing weighted-sum objective

can express weighted-sum objective as ordinary least-squares objective:

$$\begin{aligned}\|Ax-y\|^2+\mu\|Fx-g\|^2 &= \left\|\begin{bmatrix} A \\ \sqrt{\mu}F\end{bmatrix}x-\begin{bmatrix} y \\ \sqrt{\mu}g\end{bmatrix}\right\|^2 \\ &= \left\|\tilde{A}x-\tilde{y}\right\|^2\end{aligned}$$

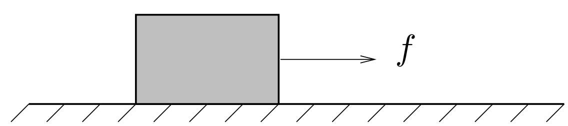
where

$$\tilde{A}=\begin{bmatrix} A \\ \sqrt{\mu}F\end{bmatrix}, \qquad \tilde{y}=\begin{bmatrix} y \\ \sqrt{\mu}g\end{bmatrix}$$

hence solution is (assuming $\tilde{A}$ full rank)

$$\begin{aligned} x &= \left(\tilde{A}^T\tilde{A}\right)^{-1}\tilde{A}^T\tilde{y} \\ &= \left(A^TA+\mu F^TF\right)^{-1}\left(A^Ty+\mu F^Tg\right)\end{aligned}$$

Example



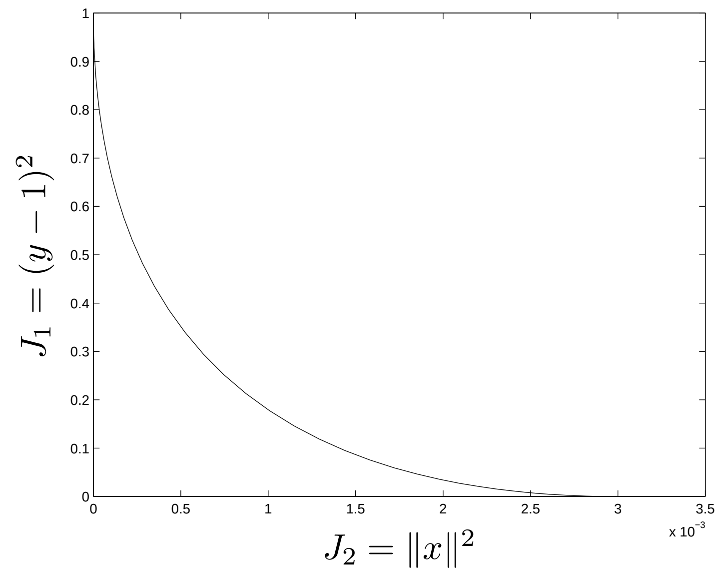
- unit mass at rest subject to forces x_i for $i-1<t\leq i$, $i=1,\ldots,10$
- $y\in\mathbf{R}$ is position at $t=10$; $y=a^Tx$ where $a\in\mathbf{R}^{10}$
- $J_1=(y-1)^2$ (final position error squared)
- $J_2=\|x\|^2$ (sum of squares of forces)

weighted-sum objective: $(a^Tx-1)^2+\mu\|x\|^2$

optimal x:

$$x=\left(aa^T+\mu I\right)^{-1}a$$

optimal trade-off curve:



- upper left corner of optimal trade-off curve corresponds to $x = 0$
- bottom right corresponds to input that yields $y = 1$, *i.e.*, $J_1 = 0$

Regularized least-squares

when $F = I$, $g = 0$ the objectives are

$$J_1 = \|Ax - y\|^2, \qquad J_2 = \|x\|^2$$

minimizer of weighted-sum objective,

$$x = \left(A^T A + \mu I\right)^{-1} A^T y,$$

is called *regularized* least-squares (approximate) solution of $Ax \approx y$

- also called *Tychonov regularization*
- for $\mu > 0$, works for *any* A (no restrictions on shape, rank . . .)

estimation/inversion application:

- $Ax - y$ is sensor residual
- prior information: x small
- or, model only accurate for x small
- regularized solution trades off sensor fit, size of x

Nonlinear least-squares

nonlinear least-squares (NLLS) problem: find $x \in \mathbf{R}^n$ that minimizes

$$\|r(x)\|^2 = \sum_{i=1}^{m} r_i(x)^2,$$

where $r : \mathbf{R}^n \to \mathbf{R}^m$

- $r(x)$ is a vector of 'residuals'
- reduces to (linear) least-squares if $r(x) = Ax - y$

Position estimation from ranges

estimate position $x \in \mathbf{R}^2$ from approximate distances to beacons at locations $b_1, \ldots, b_m \in \mathbf{R}^2$ *without* linearizing

- we measure $\rho_i = \|x - b_i\| + v_i$
  (v_i is range error, unknown but assumed small)

- NLLS estimate: choose $\hat{x}$ to minimize

$$\sum_{i=1}^{m} r_i(x)^2 = \sum_{i=1}^{m} \left(\rho_i - \|x - b_i\|\right)^2$$

Gauss-Newton method for NLLS

NLLS: find $x \in \mathbf{R}^n$ that minimizes $\|r(x)\|^2 = \sum_{i=1}^{m} r_i(x)^2$, where $r : \mathbf{R}^n \to \mathbf{R}^m$

- in general, very hard to solve exactly
- many good heuristics to compute *locally optimal* solution

Gauss-Newton method:

given starting guess for x
repeat
    linearize r near current guess
    new guess is linear LS solution, using linearized r
until convergence

Gauss-Newton method (more detail):

- linearize r near current iterate $x^{(k)}$:

$$r(x) \approx r(x^{(k)}) + Dr(x^{(k)})(x - x^{(k)})$$

  where Dr is the Jacobian: $(Dr)_{ij} = \partial r_i / \partial x_j$

- write linearized approximation as

$$r(x^{(k)}) + Dr(x^{(k)})(x - x^{(k)}) = A^{(k)}x - b^{(k)}$$

$$A^{(k)} = Dr(x^{(k)}), \qquad b^{(k)} = Dr(x^{(k)})x^{(k)} - r(x^{(k)})$$

- at kth iteration, we approximate NLLS problem by linear LS problem:

$$\|r(x)\|^2 \approx \left\| A^{(k)}x - b^{(k)} \right\|^2$$

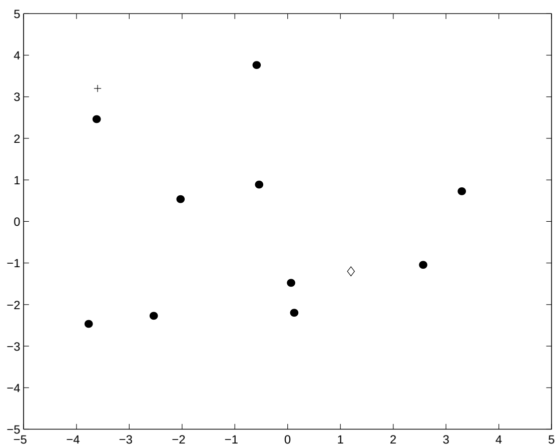
- next iterate solves this linearized LS problem:

$$x^{(k+1)} = \left(A^{(k)T} A^{(k)} \right)^{-1} A^{(k)T} b^{(k)}$$

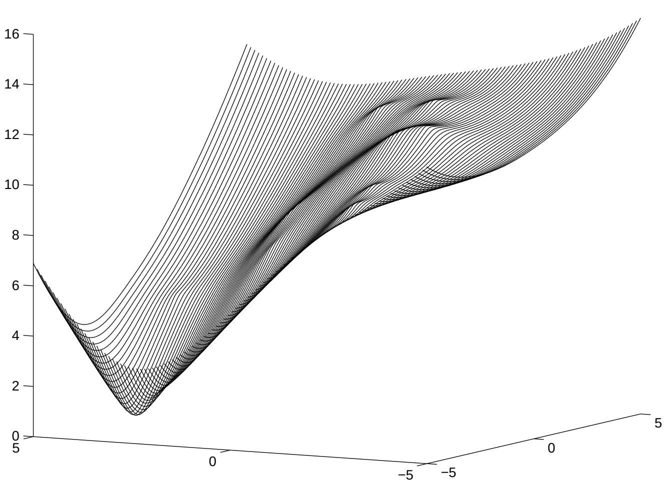
- repeat until convergence (which *isn't* guaranteed)

Gauss-Newton example

- 10 beacons
- $+$ true position $(-3.6, 3.2)$; $\diamondsuit$ initial guess $(1.2, -1.2)$
- range estimates accurate to ± 0.5

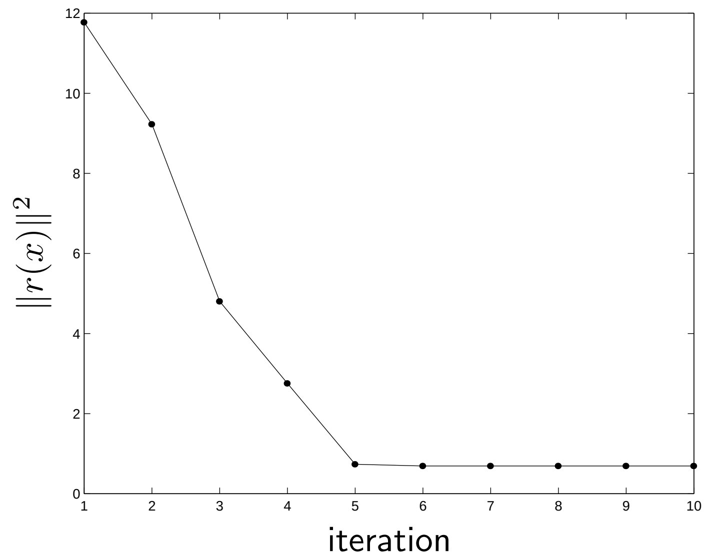


NLLS objective $\|r(x)\|^2$ versus x:



- for a linear LS problem, objective would be nice quadratic 'bowl'
- bumps in objective due to strong nonlinearity of r

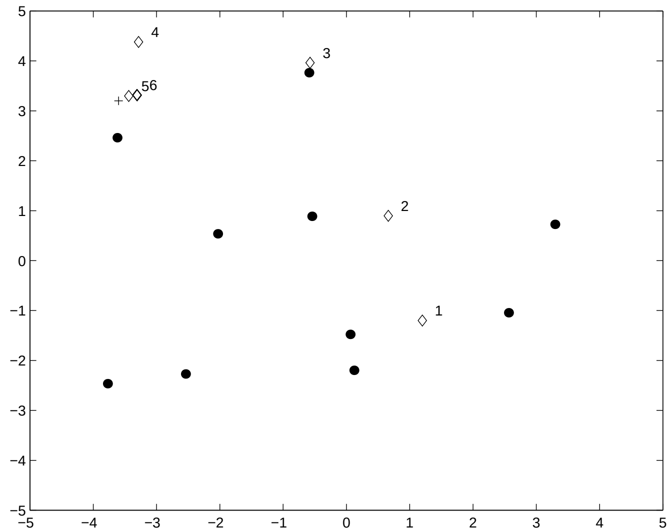
objective of Gauss-Newton iterates:



- $x^{(k)}$ converges to (in this case, global) minimum of $\|r(x)\|^2$
- convergence takes only five or so steps

- final estimate is $\hat{x} = (-3.3, 3.3)$
- estimation error is $\|\hat{x} - x\| = 0.31$
(substantially smaller than range accuracy!)

convergence of Gauss-Newton iterates:



useful varation on Gauss-Newton: add regularization term

$$\|A^{(k)}x - b^{(k)}\|^2 + \mu\|x - x^{(k)}\|^2$$

so that next iterate is not too far from previous one (hence, linearized model still pretty accurate)

Lecture 8
Least-norm solutions of undetermined equations

- least-norm solution of underdetermined equations
- minimum norm solutions via QR factorization
- derivation via Lagrange multipliers
- relation to regularized least-squares
- general norm minimization with equality constraints

Underdetermined linear equations

we consider

$$y = Ax$$

where $A \in \mathbf{R}^{m \times n}$ is fat ($m < n$), $i.e.$,

- there are more variables than equations
- x is *underspecified*, $i.e.$, many choices of x lead to the same y

we'll assume that A is full rank (m), so for each $y \in \mathbf{R}^m$, there is a solution

set of all solutions has form

$$\{ \, x \mid Ax = y \, \} = \{ \, x_p + z \mid z \in \mathcal{N}(A) \, \}$$

where x_p is any ('particular') solution, $i.e.$, $Ax_p = y$

- z characterizes available choices in solution
- solution has $\dim \mathcal{N}(A) = n - m$ 'degrees of freedom'
- can choose z to satisfy other specs or optimize among solutions

Least-norm solution

one particular solution is

$$x_{\text{ln}} = A^T(AA^T)^{-1}y$$

(AA^T is invertible since A full rank)

in fact, x_{ln} is the solution of $y = Ax$ that minimizes $\|x\|$

i.e., x_{ln} is solution of optimization problem

$$\begin{array}{ll} \text{minimize} & \|x\| \\ \text{subject to} & Ax = y \end{array}$$

(with variable $x \in \mathbf{R}^n$)

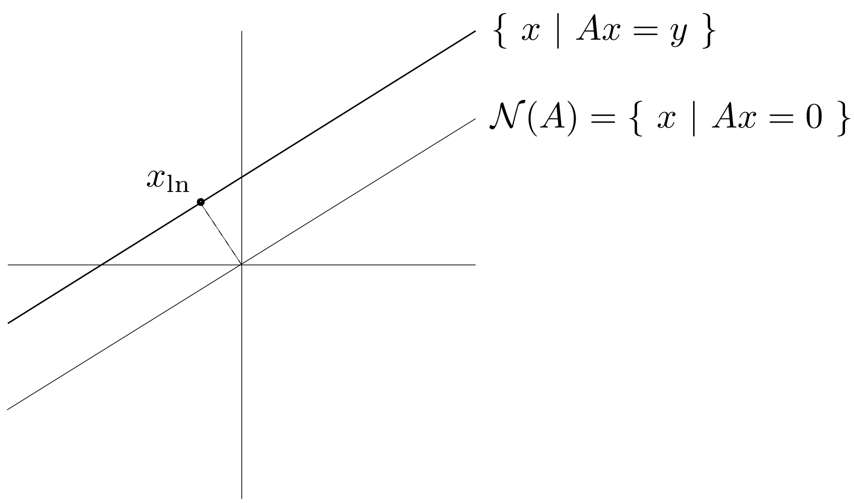
suppose $Ax = y$, so $A(x - x_{\rm ln}) = 0$ and

$$\begin{aligned}(x - x_{\rm ln})^T x_{\rm ln} &= (x - x_{\rm ln})^T A^T (AA^T)^{-1} y \\ &= (A(x - x_{\rm ln}))^T (AA^T)^{-1} y \\ &= 0\end{aligned}$$

i.e., $(x - x_{\rm ln}) \perp x_{\rm ln}$, so

$$\|x\|^2 = \|x_{\rm ln} + x - x_{\rm ln}\|^2 = \|x_{\rm ln}\|^2 + \|x - x_{\rm ln}\|^2 \geq \|x_{\rm ln}\|^2$$

i.e., $x_{\rm ln}$ has smallest norm of any solution



- **orthogonality condition:** $x_{\rm ln} \perp \mathcal{N}(A)$
- **projection interpretation:** $x_{\rm ln}$ is projection of 0 on solution set $\{ x \mid Ax = y \}$

- $A^\dagger = A^T(AA^T)^{-1}$ is called the *pseudo-inverse* of full rank, fat A
- $A^T(AA^T)^{-1}$ is a *right inverse* of A
- $I - A^T(AA^T)^{-1}A$ gives projection onto $\mathcal{N}(A)$

cf. analogous formulas for full rank, **skinny** matrix A:

- $A^\dagger = (A^TA)^{-1}A^T$
- $(A^TA)^{-1}A^T$ is a *left inverse* of A
- $A(A^TA)^{-1}A^T$ gives projection onto $\mathcal{R}(A)$

Least-norm solution via QR factorization

find QR factorization of A^T, $i.e.$, $A^T = QR$, with

- $Q \in \mathbf{R}^{n\times m}$, $Q^TQ = I_m$
- $R \in \mathbf{R}^{m\times m}$ upper triangular, nonsingular

then

- $x_{\rm ln} = A^T(AA^T)^{-1}y = QR^{-T}y$
- $\|x_{\rm ln}\| = \|R^{-T}y\|$

Derivation via Lagrange multipliers

- least-norm solution solves optimization problem

$$\begin{array}{ll} \text{minimize} & x^T x \\ \text{subject to} & Ax = y \end{array}$$

- introduce Lagrange multipliers: $L(x, \lambda) = x^T x + \lambda^T (Ax - y)$

- optimality conditions are

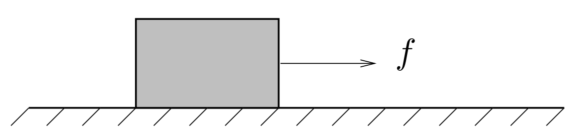
$$\nabla_x L = 2x + A^T \lambda = 0, \qquad \nabla_\lambda L = Ax - y = 0$$

- from first condition, $x = -A^T \lambda / 2$

- substitute into second to get $\lambda = -2(AA^T)^{-1} y$

- hence $x = A^T (AA^T)^{-1} y$

Example: transferring mass unit distance

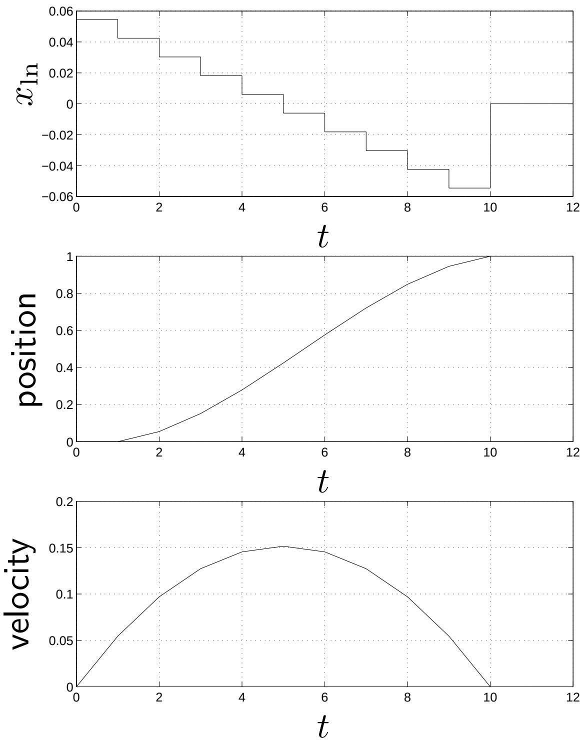


- unit mass at rest subject to forces x_i for $i - 1 < t \leq i$, $i = 1, \ldots, 10$

- y_1 is position at $t = 10$, y_2 is velocity at $t = 10$

- $y = Ax$ where $A \in \mathbf{R}^{2 \times 10}$ (A is fat)

- find least norm force that transfers mass unit distance with zero final velocity, *i.e.*, $y = (1, 0)$



Relation to regularized least-squares

- suppose $A \in \mathbf{R}^{m \times n}$ is fat, full rank
- define $J_1 = \|Ax - y\|^2$, $J_2 = \|x\|^2$
- least-norm solution minimizes J_2 with $J_1 = 0$
- minimizer of weighted-sum objective $J_1 + \mu J_2 = \|Ax - y\|^2 + \mu\|x\|^2$ is

$$x_\mu = \left(A^T A + \mu I\right)^{-1} A^T y$$

- **fact:** $x_\mu \to x_{\rm ln}$ as $\mu \to 0$, *i.e.*, regularized solution converges to least-norm solution as $\mu \to 0$
- in matrix terms: as $\mu \to 0$,

$$\left(A^T A + \mu I\right)^{-1} A^T \to A^T \left(AA^T\right)^{-1}$$

(for full rank, fat A)

General norm minimization with equality constraints

consider problem

$$\begin{array}{ll} \text{minimize} & \|Ax-b\| \\ \text{subject to} & Cx = d \end{array}$$

with variable x

- includes least-squares and least-norm problems as special cases
- equivalent to

$$\begin{array}{ll} \text{minimize} & (1/2)\|Ax-b\|^2 \\ \text{subject to} & Cx = d \end{array}$$

- Lagrangian is

$$\begin{aligned} L(x,\lambda) &= (1/2)\|Ax-b\|^2 + \lambda^T(Cx-d) \\ &= (1/2)x^TA^TAx - b^TAx + (1/2)b^Tb + \lambda^TCx - \lambda^Td \end{aligned}$$

- optimality conditions are

$$\nabla_x L = A^TAx - A^Tb + C^T\lambda = 0, \qquad \nabla_\lambda L = Cx - d = 0$$

- write in block matrix form as

$$\left[\begin{array}{cc} A^TA & C^T \\ C & 0 \end{array}\right]\left[\begin{array}{c} x \\ \lambda \end{array}\right] = \left[\begin{array}{c} A^Tb \\ d \end{array}\right]$$

- if the block matrix is invertible, we have

$$\left[\begin{array}{c} x \\ \lambda \end{array}\right] = \left[\begin{array}{cc} A^TA & C^T \\ C & 0 \end{array}\right]^{-1}\left[\begin{array}{c} A^Tb \\ d \end{array}\right]$$

if A^TA is invertible, we can derive a more explicit (and complicated) formula for x

- from first block equation we get

$$x = (A^TA)^{-1}(A^Tb - C^T\lambda)$$

- substitute into $Cx = d$ to get

$$C(A^TA)^{-1}(A^Tb - C^T\lambda) = d$$

so

$$\lambda = \left(C(A^TA)^{-1}C^T\right)^{-1}\left(C(A^TA)^{-1}A^Tb - d\right)$$

- recover x from equation above (not pretty)

$$x = (A^TA)^{-1}\left(A^Tb - C^T\left(C(A^TA)^{-1}C^T\right)^{-1}\left(C(A^TA)^{-1}A^Tb - d\right)\right)$$

Lecture 9
Autonomous linear dynamical systems

- autonomous linear dynamical systems
- examples
- higher order systems
- linearization near equilibrium point
- linearization along trajectory

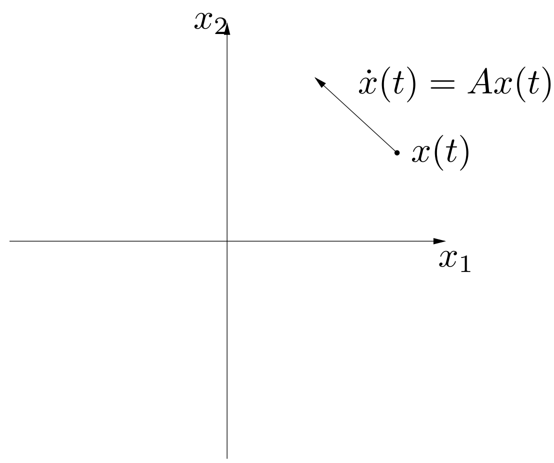
Autonomous linear dynamical systems

continuous-time autonomous LDS has form

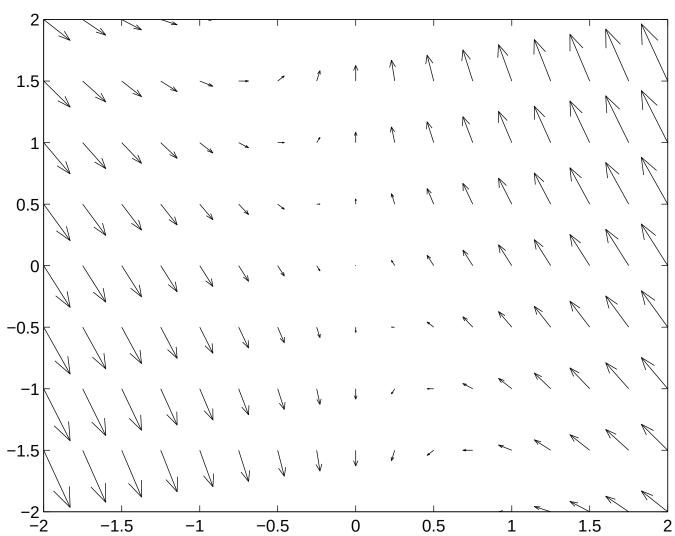
$$\dot{x} = Ax$$

- $x(t) \in \mathbf{R}^n$ is called the state
- n is the *state dimension* or (informally) the *number of states*
- A is the *dynamics matrix*
  (system is *time-invariant* if A doesn't depend on t)

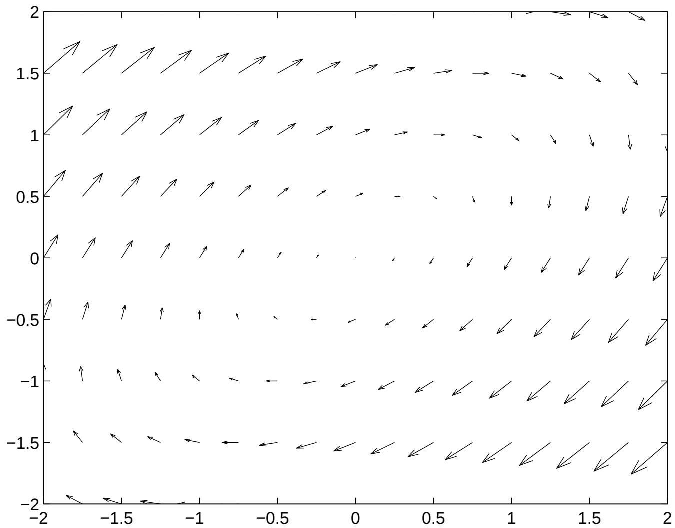
picture (*phase plane*):



example 1: $\dot{x} = \begin{bmatrix} -1 & 0 \\ 2 & 1 \end{bmatrix} x$

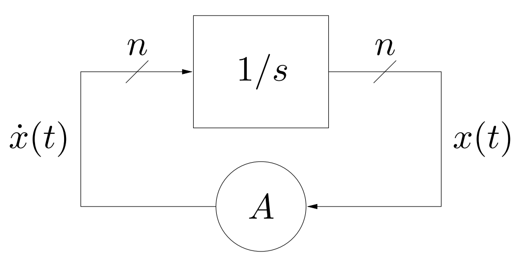


example 2: $\dot{x} = \begin{bmatrix} -0.5 & 1 \\ -1 & 0.5 \end{bmatrix} x$



Block diagram

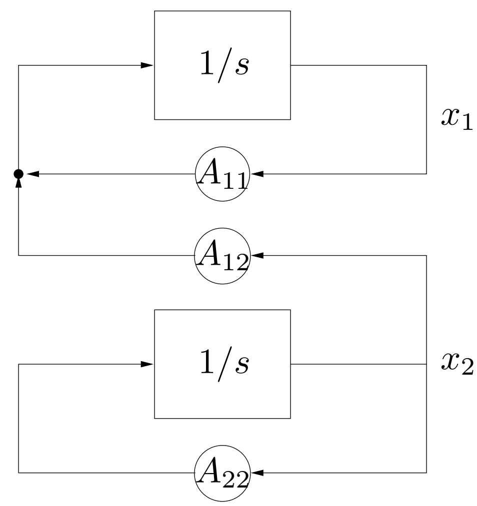
block diagram representation of $\dot{x} = Ax$:



- $1/s$ block represents n parallel scalar integrators
- coupling comes from dynamics matrix A

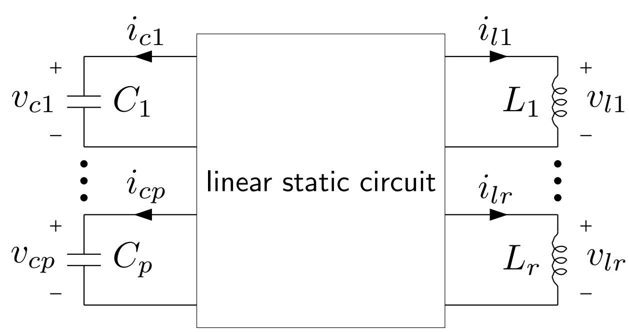
useful when A has structure, *e.g.*, block upper triangular:

$$\dot{x} = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix} x$$



here x_1 doesn't affect x_2 at all

Linear circuit



circuit equations are

$$C\frac{dv_c}{dt} = i_c, \qquad L\frac{di_l}{dt} = v_l, \qquad \begin{bmatrix} i_c \\ v_l \end{bmatrix} = F \begin{bmatrix} v_c \\ i_l \end{bmatrix}$$

$$C = \mathbf{diag}(C_1, \ldots, C_p), \qquad L = \mathbf{diag}(L_1, \ldots, L_r)$$

with state $x = \begin{bmatrix} v_c \\ i_l \end{bmatrix}$, we have

$$\dot{x} = \begin{bmatrix} C^{-1} & 0 \\ 0 & L^{-1} \end{bmatrix} Fx$$

Chemical reactions

- reaction involving n chemicals; x_i is concentration of chemical i
- linear model of reaction kinetics

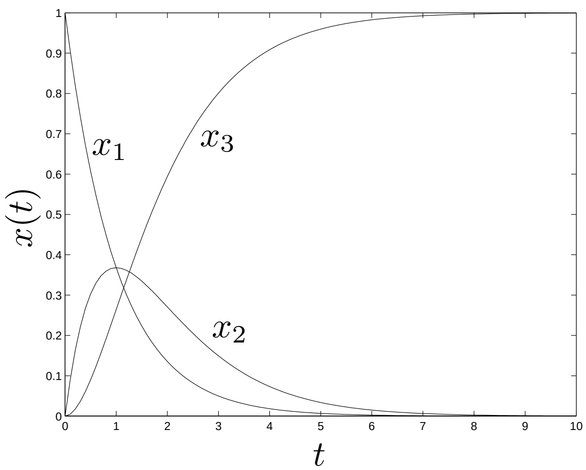
$$\frac{dx_i}{dt} = a_{i1}x_1 + \cdots + a_{in}x_n$$

- good model for some reactions; A is usually sparse

Example: series reaction $A \xrightarrow{k_1} B \xrightarrow{k_2} C$ with linear dynamics

$$\dot{x} = \begin{bmatrix} -k_1 & 0 & 0 \\ k_1 & -k_2 & 0 \\ 0 & k_2 & 0 \end{bmatrix} x$$

plot for $k_1 = k_2 = 1$, initial $x(0) = (1, 0, 0)$



Finite-state discrete-time Markov chain

$z(t) \in \{1, \ldots, n\}$ is a random sequence with

$$\mathbf{Prob}(\ z(t+1) = i \mid z(t) = j\ ) = P_{ij}$$

where $P \in \mathbf{R}^{n \times n}$ is the matrix of *transition probabilities*

can represent probability distribution of $z(t)$ as n-vector

$$p(t) = \begin{bmatrix} \mathbf{Prob}(z(t) = 1) \\ \vdots \\ \mathbf{Prob}(z(t) = n) \end{bmatrix}$$

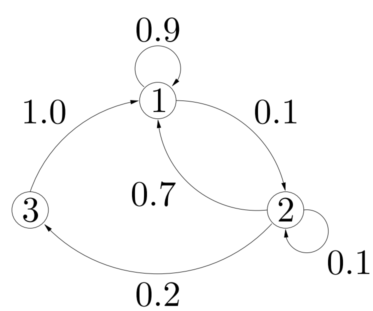
(so, $e.g.$, $\mathbf{Prob}(z(t) = 1, 2, \text{ or } 3) = [1\ 1\ 1\ 0 \cdots 0]p(t)$)

then we have $p(t+1) = Pp(t)$

P is often sparse; Markov chain is depicted graphically

- nodes are states
- edges show transition probabilities

example:



- state 1 is 'system OK'
- state 2 is 'system down'
- state 3 is 'system being repaired'

$$p(t+1) = \begin{bmatrix} 0.9 & 0.7 & 1.0 \\ 0.1 & 0.1 & 0 \\ 0 & 0.2 & 0 \end{bmatrix} p(t)$$

Numerical integration of continuous system

compute approximate solution of $\dot{x} = Ax$, $x(0) = x_0$

suppose h is small time step (x doesn't change much in h seconds)

simple ('forward Euler') approximation:

$$x(t+h) \approx x(t) + h\dot{x}(t) = (I + hA)x(t)$$

by carrying out this recursion (discrete-time LDS), starting at $x(0) = x_0$, we get approximation

$$x(kh) \approx (I + hA)^k x(0)$$

(forward Euler is never used in practice)

Higher order linear dynamical systems

$$x^{(k)} = A_{k-1}x^{(k-1)} + \cdots + A_1 x^{(1)} + A_0 x, \quad x(t) \in \mathbf{R}^n$$

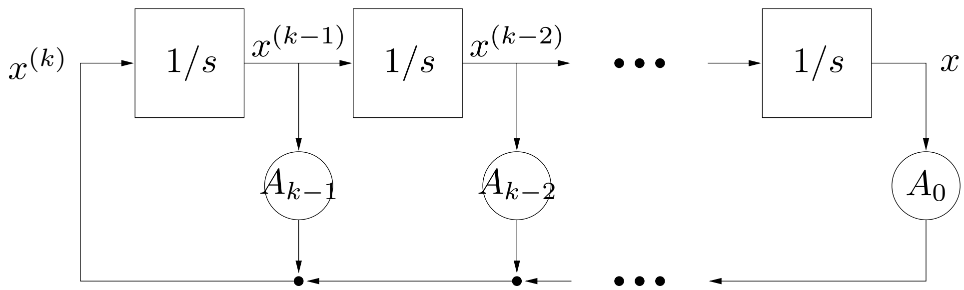
where $x^{(m)}$ denotes mth derivative

define new variable $z = \begin{bmatrix} x \\ x^{(1)} \\ \vdots \\ x^{(k-1)} \end{bmatrix} \in \mathbf{R}^{nk}$, so

$$\dot{z} = \begin{bmatrix} x^{(1)} \\ \vdots \\ x^{(k)} \end{bmatrix} = \begin{bmatrix} 0 & I & 0 & \cdots & 0 \\ 0 & 0 & I & \cdots & 0 \\ \vdots & & & & \vdots \\ 0 & 0 & 0 & \cdots & I \\ A_0 & A_1 & A_2 & \cdots & A_{k-1} \end{bmatrix} z$$

a (first order) LDS (with bigger state)

block diagram:



Mechanical systems

mechanical system with k degrees of freedom undergoing small motions:

$$M\ddot{q} + D\dot{q} + Kq = 0$$

- $q(t) \in \mathbf{R}^k$ is the vector of generalized displacements
- M is the *mass matrix*
- K is the *stiffness matrix*
- D is the *damping matrix*

with state $x = \begin{bmatrix} q \\ \dot{q} \end{bmatrix}$ we have

$$\dot{x} = \begin{bmatrix} \dot{q} \\ \ddot{q} \end{bmatrix} = \begin{bmatrix} 0 & I \\ -M^{-1}K & -M^{-1}D \end{bmatrix} x$$

Linearization near equilibrium point

nonlinear, time-invariant differential equation (DE):

$$\dot{x} = f(x)$$

where $f : \mathbf{R}^n \to \mathbf{R}^n$

suppose x_e is an *equilibrium point*, *i.e.*, $f(x_e) = 0$

(so $x(t) = x_e$ satisfies DE)

now suppose $x(t)$ is near x_e, so

$$\dot{x}(t) = f(x(t)) \approx f(x_e) + Df(x_e)(x(t) - x_e)$$

with $\delta x(t) = x(t) - x_e$, rewrite as

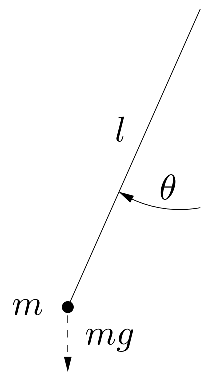
$$\dot{\delta x}(t) \approx Df(x_e)\delta x(t)$$

replacing $\approx$ with $=$ yields *linearized approximation* of DE near x_e

we *hope* solution of $\dot{\delta x} = Df(x_e)\delta x$ is a good approximation of $x - x_e$

(more later)

example: pendulum



2nd order nonlinear DE $ml^2\ddot{\theta} = -lmg\sin\theta$

rewrite as first order DE with state $x = \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix}$:

$$\dot{x} = \begin{bmatrix} x_2 \\ -(g/l)\sin x_1 \end{bmatrix}$$

equilibrium point (pendulum down): $x = 0$

linearized system near $x_e = 0$:

$$\dot{\delta x} = \begin{bmatrix} 0 & 1 \\ -g/l & 0 \end{bmatrix} \delta x$$

Does linearization 'work'?

the linearized system usually, but not always, gives a good idea of the system behavior near x_e

example 1: $\dot{x} = -x^3$ near $x_e = 0$

for $x(0) > 0$ solutions have form $x(t) = \left(x(0)^{-2} + 2t\right)^{-1/2}$

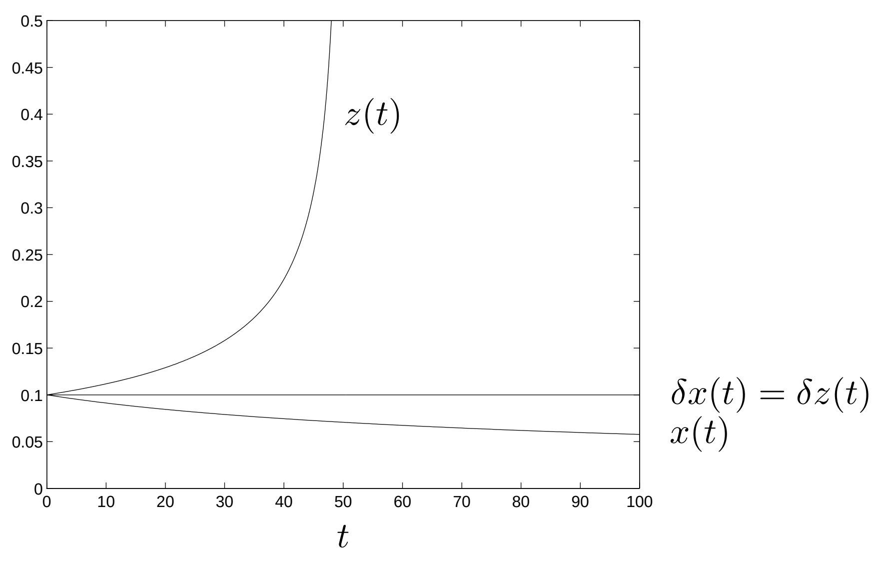
linearized system is $\dot{\delta x} = 0$; solutions are constant

example 2: $\dot{z} = z^3$ near $z_e = 0$

for $z(0) > 0$ solutions have form $z(t) = \left(z(0)^{-2} - 2t\right)^{-1/2}$

(finite escape time at $t = z(0)^{-2}/2$)

linearized system is $\dot{\delta z} = 0$; solutions are constant



- systems with very different behavior have same linearized system
- linearized systems do not predict qualitative behavior of either system

Linearization along trajectory

- suppose $x_{\rm traj} : \mathbf{R}_+ \to \mathbf{R}^n$ satisfies $\dot{x}_{\rm traj}(t) = f(x_{\rm traj}(t), t)$
- suppose $x(t)$ is another trajectory, *i.e.*, $\dot{x}(t) = f(x(t), t)$, and is near $x_{\rm traj}(t)$
- then

$$\frac{d}{dt}(x - x_{\rm traj}) = f(x, t) - f(x_{\rm traj}, t) \approx D_x f(x_{\rm traj}, t)(x - x_{\rm traj})$$

- (time-varying) LDS

$$\dot{\delta x} = D_x f(x_{\rm traj}, t)\delta x$$

  is called *linearized* or *variational system* along trajectory $x_{\rm traj}$

example: linearized oscillator

suppose $x_{\rm traj}(t)$ is T-periodic solution of nonlinear DE:

$$\dot{x}_{\rm traj}(t) = f(x_{\rm traj}(t)), \qquad x_{\rm traj}(t + T) = x_{\rm traj}(t)$$

linearized system is

$$\dot{\delta x} = A(t)\delta x$$

where $A(t) = Df(x_{\rm traj}(t))$

$A(t)$ is T-periodic, so linearized system is called *T-periodic linear system*.

used to study:

- startup dynamics of clock and oscillator circuits
- effects of power supply and other disturbances on clock behavior

Lecture 10
Solution via Laplace transform and matrix exponential

- Laplace transform
- solving $\dot{x} = Ax$ via Laplace transform
- state transition matrix
- matrix exponential
- qualitative behavior and stability

Laplace transform of matrix valued function

suppose $z : \mathbf{R}_+ \to \mathbf{R}^{p\times q}$

Laplace transform: $Z = \mathcal{L}(z)$, where $Z : D \subseteq \mathbf{C} \to \mathbf{C}^{p\times q}$ is defined by

$$Z(s) = \int_0^\infty e^{-st} z(t)\, dt$$

- integral of matrix is done term-by-term
- convention: upper case denotes Laplace transform
- D is the *domain* or *region of convergence* of Z
- D includes at least $\{s \mid \Re s > a\}$, where a satisfies $|z_{ij}(t)| \leq \alpha e^{at}$ for $t \geq 0$, $i = 1, \ldots, p$, $j = 1, \ldots, q$

Derivative property

$\mathcal{L}(\dot{z}) = sZ(s) - z(0)$

to derive, integrate by parts:

$$\begin{aligned}\mathcal{L}(\dot{z})(s) &= \int_0^\infty e^{-st}\dot{z}(t)\, dt \\ &= e^{-st}z(t)\big|_{t=0}^{t\to\infty} + s\int_0^\infty e^{-st}z(t)\, dt \\ &= sZ(s) - z(0)\end{aligned}$$

Laplace transform solution of $\dot{x} = Ax$

consider continuous-time time-invariant (TI) LDS

$$\dot{x} = Ax$$

for $t \geq 0$, where $x(t) \in \mathbf{R}^n$

- take Laplace transform: $sX(s) - x(0) = AX(s)$
- rewrite as $(sI - A)X(s) = x(0)$
- hence $X(s) = (sI - A)^{-1}x(0)$
- take inverse transform

$$x(t) = \mathcal{L}^{-1}\left((sI - A)^{-1}\right)x(0)$$

Resolvent and state transition matrix

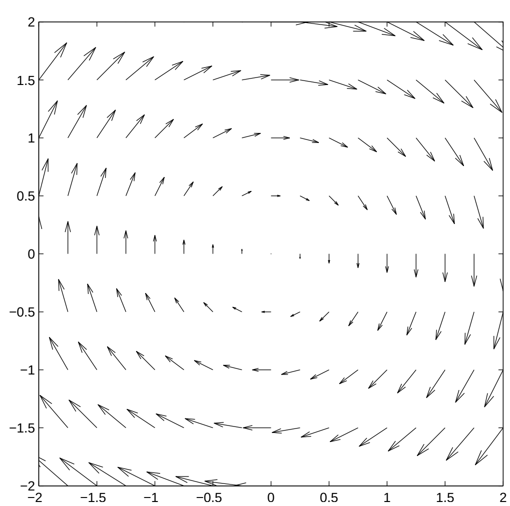
- $(sI - A)^{-1}$ is called the *resolvent* of A
- resolvent defined for $s \in \mathbf{C}$ *except* eigenvalues of A, $i.e.$, s such that $\det(sI - A) = 0$
- $\Phi(t) = \mathcal{L}^{-1}\left((sI - A)^{-1}\right)$ is called the *state-transition matrix*; it maps the initial state to the state at time t:

$$x(t) = \Phi(t)x(0)$$

(in particular, state $x(t)$ is a linear function of initial state $x(0)$)

Example 1: Harmonic oscillator

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} x$$



$sI - A = \begin{bmatrix} s & -1 \\ 1 & s \end{bmatrix}$, so resolvent is

$$(sI - A)^{-1} = \begin{bmatrix} \frac{s}{s^2+1} & \frac{1}{s^2+1} \\ \frac{-1}{s^2+1} & \frac{s}{s^2+1} \end{bmatrix}$$

(eigenvalues are $\pm j$)

state transition matrix is

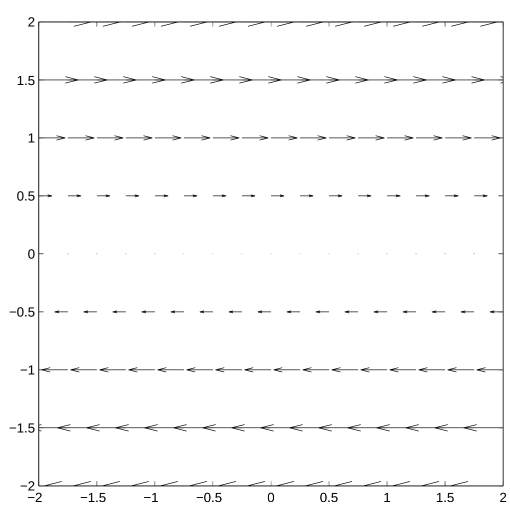
$$\Phi(t) = \mathcal{L}^{-1}\left(\begin{bmatrix} \frac{s}{s^2+1} & \frac{1}{s^2+1} \\ \frac{-1}{s^2+1} & \frac{s}{s^2+1} \end{bmatrix} \right) = \begin{bmatrix} \cos t & \sin t \\ -\sin t & \cos t \end{bmatrix}$$

a rotation matrix ($-t$ radians)

so we have $x(t) = \begin{bmatrix} \cos t & \sin t \\ -\sin t & \cos t \end{bmatrix} x(0)$

Example 2: Double integrator

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x$$



$sI - A = \begin{bmatrix} s & -1 \\ 0 & s \end{bmatrix}$, so resolvent is

$$(sI - A)^{-1} = \begin{bmatrix} \frac{1}{s} & \frac{1}{s^2} \\ 0 & \frac{1}{s} \end{bmatrix}$$

(eigenvalues are 0, 0)

state transition matrix is

$$\Phi(t) = \mathcal{L}^{-1}\left(\begin{bmatrix} \frac{1}{s} & \frac{1}{s^2} \\ 0 & \frac{1}{s} \end{bmatrix}\right) = \begin{bmatrix} 1 & t \\ 0 & 1 \end{bmatrix}$$

so we have $x(t) = \begin{bmatrix} 1 & t \\ 0 & 1 \end{bmatrix} x(0)$

Characteristic polynomial

$\mathcal{X}(s) = \det(sI - A)$ is called the *characteristic polynomial* of A

- $\mathcal{X}(s)$ is a polynomial of degree n, with leading ($i.e.$, s^n) coefficient one
- roots of $\mathcal{X}$ are the eigenvalues of A
- $\mathcal{X}$ has real coefficients, so eigenvalues are either real or occur in conjugate pairs
- there are n eigenvalues (if we count multiplicity as roots of $\mathcal{X}$)

Eigenvalues of A and poles of resolvent

i, j entry of resolvent can be expressed via Cramer's rule as

$$(-1)^{i+j} \frac{\det \Delta_{ij}}{\det(sI - A)}$$

where Δ_{ij} is $sI - A$ with jth row and ith column deleted

- $\det \Delta_{ij}$ is a polynomial of degree less than n, so i, j entry of resolvent has form $f_{ij}(s)/\mathcal{X}(s)$ where f_{ij} is polynomial with degree less than n
- poles of entries of resolvent must be eigenvalues of A
- but not all eigenvalues of A show up as poles of each entry (when there are cancellations between $\det \Delta_{ij}$ and $\mathcal{X}(s)$)

Matrix exponential

$(I - C)^{-1} = I + C + C^2 + C^3 + \cdots$ (if series converges)

- series expansion of resolvent:

$$(sI - A)^{-1} = (1/s)(I - A/s)^{-1} = \frac{I}{s} + \frac{A}{s^2} + \frac{A^2}{s^3} + \cdots$$

(valid for $|s|$ large enough) so

$$\Phi(t) = \mathcal{L}^{-1}\left((sI - A)^{-1}\right) = I + tA + \frac{(tA)^2}{2!} + \cdots$$

- looks like ordinary power series

$$e^{at} = 1 + ta + \frac{(ta)^2}{2!} + \cdots$$

with square matrices instead of scalars . . .

- define **matrix exponential** as

$$e^{M} = I + M + \frac{M^2}{2!} + \cdots$$

for $M \in \mathbf{R}^{n\times n}$ (which in fact converges for all M)

- with this definition, state-transition matrix is

$$\Phi(t) = \mathcal{L}^{-1}\left((sI - A)^{-1}\right) = e^{tA}$$

Matrix exponential solution of autonomous LDS

solution of $\dot{x} = Ax$, with $A \in \mathbf{R}^{n\times n}$ and constant, is

$$x(t) = e^{tA}x(0)$$

generalizes scalar case: solution of $\dot{x} = ax$, with $a \in \mathbf{R}$ and constant, is

$$x(t) = e^{ta}x(0)$$

- matrix exponential is *meant* to look like scalar exponential

- some things you'd guess hold for the matrix exponential (by analogy with the scalar exponential) do in fact hold

- but **many things you'd guess are wrong**

example: you might guess that $e^{A+B} = e^A e^B$, but it's false (in general)

$$A = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}, \qquad B = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$$

$$e^A = \begin{bmatrix} 0.54 & 0.84 \\ -0.84 & 0.54 \end{bmatrix}, \qquad e^B = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$

$$e^{A+B} = \begin{bmatrix} 0.16 & 1.40 \\ -0.70 & 0.16 \end{bmatrix} \neq e^A e^B = \begin{bmatrix} 0.54 & 1.38 \\ -0.84 & -0.30 \end{bmatrix}$$

however, we do have $e^{A+B} = e^A e^B$ if $AB = BA$, *i.e.*, A and B commute

thus for $t,\ s \in \mathbf{R}$, $e^{(tA+sA)} = e^{tA} e^{sA}$

with $s = -t$ we get

$$e^{tA} e^{-tA} = e^{tA-tA} = e^0 = I$$

so e^{tA} is nonsingular, with inverse

$$\left(e^{tA}\right)^{-1} = e^{-tA}$$

example: let's find e^A, where $A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$

we already found

$$e^{tA} = \mathcal{L}^{-1}(sI - A)^{-1} = \begin{bmatrix} 1 & t \\ 0 & 1 \end{bmatrix}$$

so, plugging in $t = 1$, we get $e^A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$

let's check power series:

$$e^A = I + A + \frac{A^2}{2!} + \cdots = I + A$$

since $A^2 = A^3 = \cdots = 0$

Time transfer property

for $\dot{x} = Ax$ we know

$$x(t) = \Phi(t)x(0) = e^{tA}x(0)$$

interpretation: the matrix e^{tA} propagates initial condition into state at time t

more generally we have, for *any* t and τ,

$$x(\tau + t) = e^{tA}x(\tau)$$

(to see this, apply result above to $z(t) = x(t + \tau)$)

interpretation: the matrix e^{tA} propagates state t seconds forward in time (backward if $t < 0$)

- recall first order (forward Euler) *approximate* state update, for small t:

$$x(\tau + t) \approx x(\tau) + t\dot{x}(\tau) = (I + tA)x(\tau)$$

- *exact* solution is

$$x(\tau + t) = e^{tA}x(\tau) = (I + tA + (tA)^2/2! + \cdots)x(\tau)$$

- forward Euler is just first two terms in series

Sampling a continuous-time system

suppose $\dot{x} = Ax$

sample x at times $t_1 \leq t_2 \leq \cdots$: define $z(k) = x(t_k)$

then $z(k+1) = e^{(t_{k+1} - t_k)A} z(k)$

for uniform sampling $t_{k+1} - t_k = h$, so

$$z(k+1) = e^{hA} z(k),$$

a discrete-time LDS (called *discretized version* of continuous-time system)

Piecewise constant system

consider *time-varying* LDS $\dot{x} = A(t)x$, with

$$A(t) = \begin{cases} A_0 & 0 \leq t < t_1 \\ A_1 & t_1 \leq t < t_2 \\ \vdots & \end{cases}$$

where $0 < t_1 < t_2 < \cdots$ (sometimes called jump linear system)

for $t \in [t_i, t_{i+1}]$ we have

$$x(t) = e^{(t-t_i)A_i} \cdots e^{(t_3-t_2)A_2} e^{(t_2-t_1)A_1} e^{t_1 A_0} x(0)$$

(matrix on righthand side is called state transition matrix for system, and denoted $\Phi(t)$)

Qualitative behavior of $x(t)$

suppose $\dot{x} = Ax$, $x(t) \in \mathbf{R}^n$

then $x(t) = e^{tA}x(0)$; $X(s) = (sI - A)^{-1}x(0)$

ith component $X_i(s)$ has form

$$X_i(s) = \frac{a_i(s)}{\mathcal{X}(s)}$$

where a_i is a polynomial of degree $< n$

thus the poles of X_i are all eigenvalues of A (but not necessarily the other way around) first assume eigenvalues λ_i are distinct, so $X_i(s)$ cannot have repeated poles

then $x_i(t)$ has form

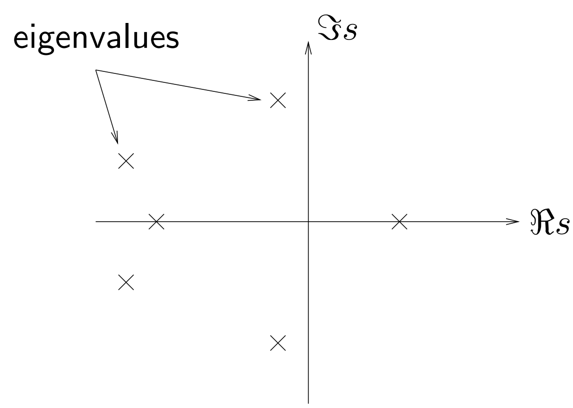
$$x_i(t) = \sum_{j=1}^{n} \beta_{ij} e^{\lambda_j t}$$

where β_{ij} depend on $x(0)$ (linearly)

eigenvalues determine (possible) qualitative behavior of x:

- eigenvalues give exponents that can occur in exponentials
- real eigenvalue λ corresponds to an exponentially decaying or growing term $e^{\lambda t}$ in solution
- complex eigenvalue $\lambda = \sigma + j\omega$ corresponds to decaying or growing sinusoidal term $e^{\sigma t}\cos(\omega t + \phi)$ in solution

- $\Re\lambda_j$ gives exponential growth rate (if > 0), or exponential decay rate (if < 0) of term
- $\Im\lambda_j$ gives frequency of oscillatory term (if $\neq 0$)



now suppose A has repeated eigenvalues, so X_i can have repeated poles

express eigenvalues as $\lambda_1, \ldots, \lambda_r$ (distinct) with multiplicities $n_1, \ldots, n_r$, respectively ($n_1 + \cdots + n_r = n$)

then $x_i(t)$ has form

$$x_i(t) = \sum_{j=1}^{r} p_{ij}(t) e^{\lambda_j t}$$

where $p_{ij}(t)$ is a polynomial of degree $< n_j$ (that depends linearly on $x(0)$)

Stability

we say system $\dot{x} = Ax$ is *stable* if $e^{tA} \to 0$ as $t \to \infty$

meaning:

- state $x(t)$ converges to 0, as $t \to \infty$, no matter what $x(0)$ is
- all trajectories of $\dot{x} = Ax$ converge to 0 as $t \to \infty$

fact: $\dot{x} = Ax$ is stable if and only if all eigenvalues of A have negative real part:

$$\Re \lambda_i < 0, \quad i = 1, \ldots, n$$

the 'if' part is clear since

$$\lim_{t\to\infty} p(t)e^{\lambda t} = 0$$

for any polynomial, if $\Re\lambda < 0$

we'll see the 'only if' part next lecture

more generally, $\max_i \Re\lambda_i$ determines the maximum asymptotic logarithmic growth rate of $x(t)$ (or decay, if < 0)

Lecture 11
Eigenvectors and diagonalization

- eigenvectors
- dynamic interpretation: invariant sets
- complex eigenvectors & invariant planes
- left eigenvectors
- diagonalization
- modal form
- discrete-time stability

Eigenvectors and eigenvalues

$\lambda \in \mathbf{C}$ is an *eigenvalue* of $A \in \mathbf{C}^{n\times n}$ if

$$\mathcal{X}(\lambda) = \det(\lambda I - A) = 0$$

equivalent to:

- there exists nonzero $v \in \mathbf{C}^n$ s.t. $(\lambda I - A)v = 0$, *i.e.*,

  $$Av = \lambda v$$

  any such v is called an *eigenvector* of A (associated with eigenvalue λ)

- there exists nonzero $w \in \mathbf{C}^n$ s.t. $w^T(\lambda I - A) = 0$, *i.e.*,

  $$w^T A = \lambda w^T$$

  any such w is called a *left eigenvector* of A

- if v is an eigenvector of A with eigenvalue λ, then so is αv, for any $\alpha \in \mathbf{C}$, $\alpha \neq 0$

- even when A is real, eigenvalue λ and eigenvector v can be complex

- when A and λ are real, we can always find a real eigenvector v associated with λ: if $Av = \lambda v$, with $A \in \mathbf{R}^{n \times n}$, $\lambda \in \mathbf{R}$, and $v \in \mathbf{C}^n$, then
$$A\Re v = \lambda \Re v, \qquad A\Im v = \lambda \Im v$$
so $\Re v$ and $\Im v$ are real eigenvectors, if they are nonzero
(and at least one is)

- *conjugate symmetry*: if A is real and $v \in \mathbf{C}^n$ is an eigenvector associated with $\lambda \in \mathbf{C}$, then $\overline{v}$ is an eigenvector associated with $\overline{\lambda}$:

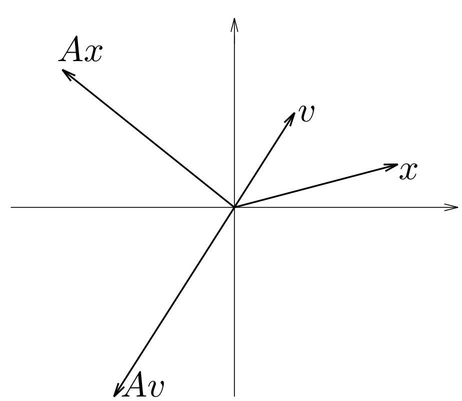
  taking conjugate of $Av = \lambda v$ we get $\overline{Av} = \overline{\lambda v}$, so
$$A\overline{v} = \overline{\lambda}\overline{v}$$

we'll assume A is real from now on . . .

Scaling interpretation

(assume $\lambda \in \mathbf{R}$ for now; we'll consider $\lambda \in \mathbf{C}$ later)

if v is an eigenvector, effect of A on v is very simple: scaling by λ



(what is λ here?)

- $\lambda \in \mathbf{R}$, $\lambda > 0$: v and Av point in same direction
- $\lambda \in \mathbf{R}$, $\lambda < 0$: v and Av point in opposite directions
- $\lambda \in \mathbf{R}$, $|\lambda| < 1$: Av smaller than v
- $\lambda \in \mathbf{R}$, $|\lambda| > 1$: Av larger than v

(we'll see later how this relates to stability of continuous- and discrete-time systems. . .)

Dynamic interpretation

suppose $Av = \lambda v$, $v \neq 0$

if $\dot{x} = Ax$ and $x(0) = v$, then $x(t) = e^{\lambda t} v$

several ways to see this, *e.g.*,

$$\begin{aligned} x(t) = e^{tA} v &= \left(I + tA + \frac{(tA)^2}{2!} + \cdots \right) v \\ &= v + \lambda t v + \frac{(\lambda t)^2}{2!} v + \cdots \\ &= e^{\lambda t} v \end{aligned}$$

(since $(tA)^k v = (\lambda t)^k v$)

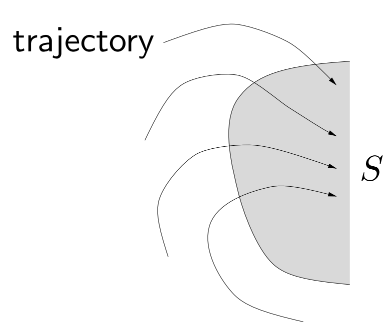
- for $\lambda \in \mathbf{C}$, solution is complex (we'll interpret later); for now, assume $\lambda \in \mathbf{R}$
- if initial state is an eigenvector v, resulting motion is very simple — always on the line spanned by v
- solution $x(t) = e^{\lambda t} v$ is called *mode* of system $\dot{x} = Ax$ (associated with eigenvalue λ)

- for $\lambda \in \mathbf{R}$, $\lambda < 0$, mode contracts or shrinks as $t \uparrow$
- for $\lambda \in \mathbf{R}$, $\lambda > 0$, mode expands or grows as $t \uparrow$

Invariant sets

a set $S \subseteq \mathbf{R}^n$ is *invariant* under $\dot{x} = Ax$ if whenever $x(t) \in S$, then $x(\tau) \in S$ for all $\tau \geq t$

$i.e.$: once trajectory enters S, it stays in S



vector field interpretation: trajectories only cut *into* S, never out suppose $Av = \lambda v$, $v \neq 0$, $\lambda \in \mathbf{R}$

- line $\{ tv \mid t \in \mathbf{R} \}$ is invariant
  (in fact, ray $\{ tv \mid t > 0 \}$ is invariant)
- if $\lambda < 0$, line segment $\{ tv \mid 0 \leq t \leq a \}$ is invariant

Complex eigenvectors

suppose $Av = \lambda v$, $v \neq 0$, λ is complex

for $a \in \mathbf{C}$, (complex) trajectory $ae^{\lambda t}v$ satisfies $\dot{x} = Ax$

hence so does (real) trajectory

$$\begin{aligned} x(t) &= \Re\left(ae^{\lambda t}v\right) \\ &= e^{\sigma t} \left[\begin{array}{cc} v_{\mathrm{re}} & v_{\mathrm{im}} \end{array}\right] \left[\begin{array}{cc} \cos\omega t & \sin\omega t \\ -\sin\omega t & \cos\omega t \end{array}\right] \left[\begin{array}{c} \alpha \\ -\beta \end{array}\right] \end{aligned}$$

where

$$v = v_{\mathrm{re}} + jv_{\mathrm{im}}, \quad \lambda = \sigma + j\omega, \quad a = \alpha + j\beta$$

- trajectory stays in *invariant plane* $\mathrm{span}\{v_{\mathrm{re}}, v_{\mathrm{im}}\}$
- σ gives logarithmic growth/decay factor
- ω gives angular velocity of rotation in plane

Dynamic interpretation: left eigenvectors

suppose $w^T A = \lambda w^T$, $w \neq 0$

then

$$\frac{d}{dt}(w^T x) = w^T \dot{x} = w^T A x = \lambda (w^T x)$$

$i.e.$, $w^T x$ satisfies the DE $d(w^T x)/dt = \lambda (w^T x)$

hence $w^T x(t) = e^{\lambda t} w^T x(0)$

- even if trajectory x is complicated, $w^T x$ is simple
- if, $e.g.$, $\lambda \in \mathbf{R}$, $\lambda < 0$, halfspace $\{ z \mid w^T z \leq a \}$ is invariant (for $a \geq 0$)
- for $\lambda = \sigma + j\omega \in \mathbf{C}$, $(\Re w)^T x$ and $(\Im w)^T x$ both have form

$$e^{\sigma t} \left(\alpha \cos(\omega t) + \beta \sin(\omega t) \right)$$

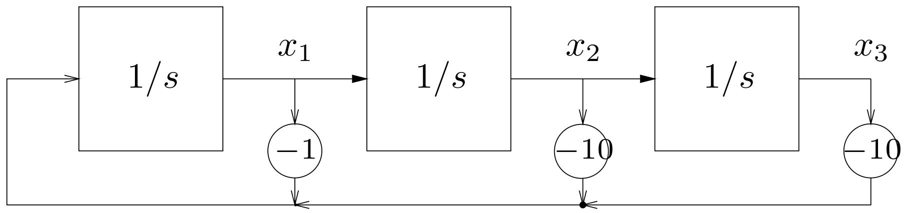
Summary

- *right eigenvectors* are initial conditions from which resulting motion is simple ($i.e.$, remains on line or in plane)

- *left eigenvectors* give linear functions of state that are simple, for any initial condition

example 1: $\dot{x} = \begin{bmatrix} -1 & -10 & -10 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} x$

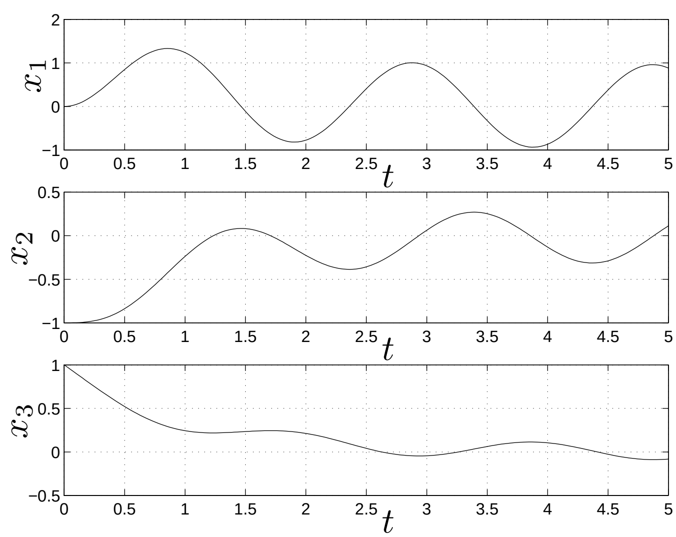
block diagram:



$\mathcal{X}(s) = s^3 + s^2 + 10s + 10 = (s+1)(s^2+10)$

eigenvalues are $-1, \ \pm j\sqrt{10}$

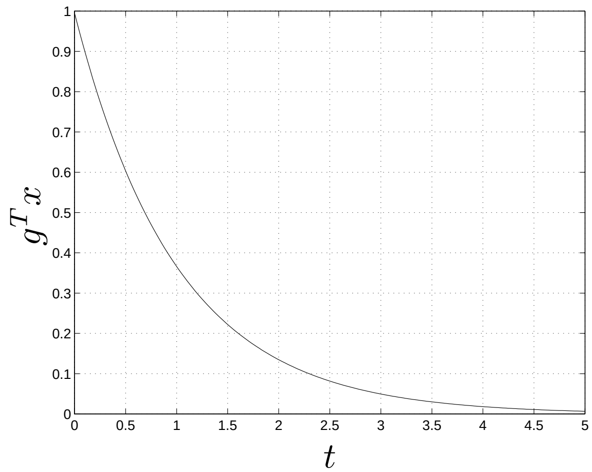
trajectory with $x(0) = (0, -1, 1)$:



left eigenvector asssociated with eigenvalue -1 is

$$g = \begin{bmatrix} 0.1 \\ 0 \\ 1 \end{bmatrix}$$

let's check $g^T x(t)$ when $x(0) = (0, -1, 1)$ (as above):



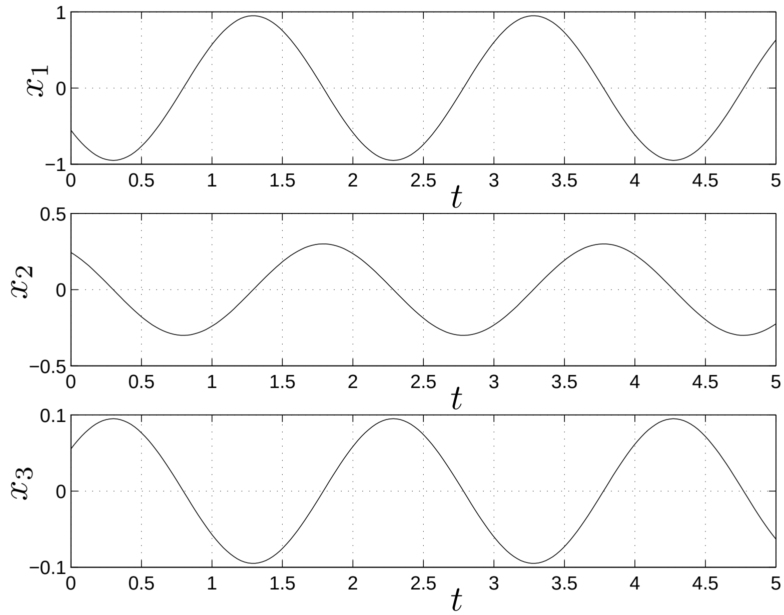
eigenvector associated with eigenvalue $j\sqrt{10}$ is

$$v = \begin{bmatrix} -0.554 + j0.771 \\ 0.244 + j0.175 \\ 0.055 - j0.077 \end{bmatrix}$$

so an invariant plane is spanned by

$$v_{\rm re} = \begin{bmatrix} -0.554 \\ 0.244 \\ 0.055 \end{bmatrix}, \quad v_{\rm im} = \begin{bmatrix} 0.771 \\ 0.175 \\ -0.077 \end{bmatrix}$$

for example, with $x(0) = v_{\mathrm{re}}$ we have



Example 2: Markov chain

probability distribution satisfies $p(t+1) = Pp(t)$

$p_i(t) = \mathbf{Prob}(\ z(t) = i\ )$ so $\sum_{i=1}^n p_i(t) = 1$

$P_{ij} = \mathbf{Prob}(\ z(t+1) = i \mid z(t) = j\ )$, so $\sum_{i=1}^n P_{ij} = 1$
(such matrices are called *stochastic*)

rewrite as:

$$[1\ 1\ \cdots\ 1]P = [1\ 1\ \cdots\ 1]$$

$i.e.$, $[1\ 1\ \cdots\ 1]$ is a left eigenvector of P with e.v. 1

hence $\det(I - P) = 0$, so there is a right eigenvector $v \neq 0$ with $Pv = v$

it can be shown that v can be chosen so that $v_i \geq 0$, hence we can normalize v so that $\sum_{i=1}^n v_i = 1$

interpretation: v is an *equilibrium distribution*; $i.e.$, if $p(0) = v$ then $p(t) = v$ for all $t \geq 0$

(if v is unique it is called the *steady-state distribution* of the Markov chain)

Diagonalization

suppose $v_1, \ldots, v_n$ is a *linearly independent* set of eigenvectors of $A \in \mathbf{R}^{n \times n}$:

$$Av_i = \lambda_i v_i, \quad i = 1, \ldots, n$$

express as

$$A \left[\begin{array}{ccc} v_1 & \cdots & v_n \end{array} \right] = \left[\begin{array}{ccc} v_1 & \cdots & v_n \end{array} \right] \left[\begin{array}{ccc} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{array} \right]$$

define $T = \left[\begin{array}{ccc} v_1 & \cdots & v_n \end{array} \right]$ and $\Lambda = \mathbf{diag}(\lambda_1, \ldots, \lambda_n)$, so

$$AT = T\Lambda$$

and finally

$$T^{-1}AT = \Lambda$$

- T invertible since $v_1, \ldots, v_n$ linearly independent
- similarity transformation by T *diagonalizes* A

conversely if there is a $T = [v_1 \ \cdots \ v_n]$ s.t.

$$T^{-1}AT = \Lambda = \mathbf{diag}(\lambda_1, \ldots, \lambda_n)$$

then $AT = T\Lambda$, $i.e.$,

$$Av_i = \lambda_i v_i, \quad i = 1, \ldots, n$$

so $v_1, \ldots, v_n$ is a linearly independent set of eigenvectors of A

we say A is *diagonalizable* if

- there exists T s.t. $T^{-1}AT = \Lambda$ is diagonal
- A has a set of linearly independent eigenvectors

(if A is not diagonalizable, it is sometimes called *defective*)

Not all matrices are diagonalizable

example: $A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$

characteristic polynomial is $\mathcal{X}(s) = s^2$, so $\lambda = 0$ is only eigenvalue

eigenvectors satisfy $Av = 0v = 0$, $i.e.$

$$\begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = 0$$

so all eigenvectors have form $v = \begin{bmatrix} v_1 \\ 0 \end{bmatrix}$ where $v_1 \neq 0$

thus, A cannot have two independent eigenvectors

Distinct eigenvalues

fact: if A has distinct eigenvalues, $i.e.$, $\lambda_i \neq \lambda_j$ for $i \neq j$, then A is diagonalizable

(the converse is false — A can have repeated eigenvalues but still be diagonalizable)

Diagonalization and left eigenvectors

rewrite $T^{-1}AT = \Lambda$ as $T^{-1}A = \Lambda T^{-1}$, or

$$\begin{bmatrix} w_1^T \\ \vdots \\ w_n^T \end{bmatrix} A = \Lambda \begin{bmatrix} w_1^T \\ \vdots \\ w_n^T \end{bmatrix}$$

where $w_1^T, \ldots, w_n^T$ are the rows of T^{-1}

thus

$$w_i^T A = \lambda_i w_i^T$$

$i.e.$, the rows of T^{-1} are (lin. indep.) left eigenvectors, normalized so that

$$w_i^T v_j = \delta_{ij}$$

($i.e.$, left & right eigenvectors chosen this way are *dual bases*)

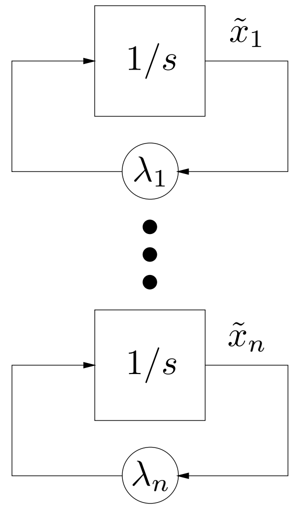
Modal form

suppose A is diagonalizable by T

define new coordinates by $x = T\tilde{x}$, so

$$T\dot{\tilde{x}} = AT\tilde{x} \quad \Leftrightarrow \quad \dot{\tilde{x}} = T^{-1}AT\tilde{x} \quad \Leftrightarrow \quad \dot{\tilde{x}} = \Lambda\tilde{x}$$

in new coordinate system, system is diagonal (decoupled):



trajectories consist of n independent modes, $i.e.$,

$$\tilde{x}_i(t) = e^{\lambda_i t}\tilde{x}_i(0)$$

hence the name *modal form*

Real modal form

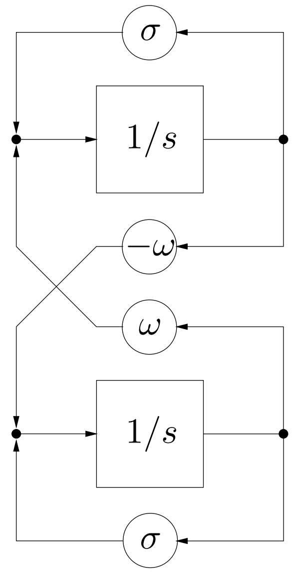
when eigenvalues (hence T) are complex, system can be put in *real modal form*:

$$S^{-1}AS = \mathbf{diag}\left(\Lambda_r, M_{r+1}, M_{r+3}, \ldots, M_{n-1}\right)$$

where $\Lambda_r = \mathbf{diag}(\lambda_1, \ldots, \lambda_r)$ are the real eigenvalues, and

$$M_i = \begin{bmatrix} \sigma_i & \omega_i \\ -\omega_i & \sigma_i \end{bmatrix}, \quad \lambda_i = \sigma_i + j\omega_i, \quad i = r+1, r+3, \ldots, n$$

where λ_i are the complex eigenvalues (one from each conjugate pair) block diagram of 'complex mode':



diagonalization simplifies many matrix expressions

e.g., resolvent:

$$\begin{aligned}(sI-A)^{-1} &= \left(sTT^{-1}-T\Lambda T^{-1}\right)^{-1}\\ &= \left(T(sI-\Lambda)T^{-1}\right)^{-1}\\ &= T(sI-\Lambda)^{-1}T^{-1}\\ &= T\,\mathbf{diag}\left(\frac{1}{s-\lambda_1},\ldots,\frac{1}{s-\lambda_n}\right)T^{-1}\end{aligned}$$

powers (*i.e.*, discrete-time solution):

$$\begin{aligned}A^k &= \left(T\Lambda T^{-1}\right)^k\\ &= \left(T\Lambda T^{-1}\right)\cdots\left(T\Lambda T^{-1}\right)\\ &= T\Lambda^k T^{-1}\\ &= T\,\mathbf{diag}(\lambda_1^k,\ldots,\lambda_n^k)T^{-1}\end{aligned}$$

(for $k<0$ only if A invertible, *i.e.*, all $\lambda_i \neq 0$)

exponential ($i.e.$, continuous-time solution):

$$\begin{aligned} e^A &= I + A + A^2/2! + \cdots \\ &= I + T\Lambda T^{-1} + \left(T\Lambda T^{-1}\right)^2/2! + \cdots \\ &= T(I + \Lambda + \Lambda^2/2! + \cdots)T^{-1} \\ &= Te^{\Lambda}T^{-1} \\ &= T\,\mathbf{diag}(e^{\lambda_1}, \ldots, e^{\lambda_n})T^{-1} \end{aligned}$$

Analytic function of a matrix

for any analytic function $f : \mathbf{R} \to \mathbf{R}$, $i.e.$, given by power series

$$f(a) = \beta_0 + \beta_1 a + \beta_2 a^2 + \beta_3 a^3 + \cdots$$

we can define $f(A)$ for $A \in \mathbf{R}^{n\times n}$ ($i.e.$, overload f) as

$$f(A) = \beta_0 I + \beta_1 A + \beta_2 A^2 + \beta_3 A^3 + \cdots$$

substituting $A = T\Lambda T^{-1}$, we have

$$\begin{aligned} f(A) &= \beta_0 I + \beta_1 A + \beta_2 A^2 + \beta_3 A^3 + \cdots \\ &= \beta_0 TT^{-1} + \beta_1 T\Lambda T^{-1} + \beta_2 (T\Lambda T^{-1})^2 + \cdots \\ &= T\left(\beta_0 I + \beta_1 \Lambda + \beta_2 \Lambda^2 + \cdots\right)T^{-1} \\ &= T\,\mathbf{diag}(f(\lambda_1), \ldots, f(\lambda_n))T^{-1} \end{aligned}$$

Solution via diagonalization

assume A is diagonalizable

consider LDS $\dot{x} = Ax$, with $T^{-1}AT = \Lambda$

then

$$\begin{aligned} x(t) &= e^{tA}x(0) \\ &= Te^{\Lambda t}T^{-1}x(0) \\ &= \sum_{i=1}^{n} e^{\lambda_i t}(w_i^T x(0))v_i \end{aligned}$$

thus: any trajectory can be expressed as linear combination of modes

interpretation:

- (left eigenvectors) decompose initial state $x(0)$ into modal components $w_i^T x(0)$
- $e^{\lambda_i t}$ term propagates ith mode forward t seconds
- reconstruct state as linear combination of (right) eigenvectors

application: for what $x(0)$ do we have $x(t) \to 0$ as $t \to \infty$?

divide eigenvalues into those with negative real parts

$$\Re\lambda_1 < 0, \ldots, \Re\lambda_s < 0,$$

and the others,

$$\Re\lambda_{s+1} \geq 0, \ldots, \Re\lambda_n \geq 0$$

from

$$x(t) = \sum_{i=1}^{n} e^{\lambda_i t}(w_i^T x(0)) v_i$$

condition for $x(t) \to 0$ is:

$$x(0) \in \text{span}\{v_1, \ldots, v_s\},$$

or equivalently,

$$w_i^T x(0) = 0, \quad i = s+1, \ldots, n$$

(can you prove this?)

Stability of discrete-time systems

suppose A diagonalizable

consider discrete-time LDS $x(t+1) = Ax(t)$

if $A = T\Lambda T^{-1}$, then $A^k = T\Lambda^k T^{-1}$

then

$$x(t) = A^t x(0) = \sum_{i=1}^{n} \lambda_i^t (w_i^T x(0)) v_i \to 0 \quad \text{as } t \to \infty$$

for all $x(0)$ if and only if

$$|\lambda_i| < 1, \quad i = 1, \ldots, n.$$

we will see later that this is true even when A is not diagonalizable, so we have

fact: $x(t+1) = Ax(t)$ is stable if and only if all eigenvalues of A have magnitude less than one

Lecture 12
Jordan canonical form

- Jordan canonical form
- generalized modes
- Cayley-Hamilton theorem

Jordan canonical form

what if A cannot be diagonalized?

any matrix $A \in \mathbf{R}^{n\times n}$ can be put in *Jordan canonical form* by a similarity transformation, $i.e.$

$$T^{-1}AT = J = \begin{bmatrix} J_1 & & \\ & \ddots & \\ & & J_q \end{bmatrix}$$

where

$$J_i = \begin{bmatrix} \lambda_i & 1 & & \\ & \lambda_i & \ddots & \\ & & \ddots & 1 \\ & & & \lambda_i \end{bmatrix} \in \mathbf{C}^{n_i\times n_i}$$

is called a *Jordan block* of size n_i with eigenvalue λ_i (so $n = \sum_{i=1}^{q} n_i$)

- J is upper bidiagonal
- J diagonal is the special case of n Jordan blocks of size $n_i = 1$
- Jordan form is unique (up to permutations of the blocks)
- can have multiple blocks with same eigenvalue

note: JCF is a *conceptual tool*, never used in numerical computations!

$\mathcal{X}(s) = \det(sI - A) = (s - \lambda_1)^{n_1} \cdots (s - \lambda_q)^{n_q}$

hence distinct eigenvalues $\Rightarrow n_i = 1 \Rightarrow A$ diagonalizable

$\dim \mathcal{N}(\lambda I - A)$ is the number of Jordan blocks with eigenvalue λ

more generally,

$$\dim \mathcal{N}(\lambda I - A)^k = \sum_{\lambda_i = \lambda} \min\{k, n_i\}$$

so from $\dim \mathcal{N}(\lambda I - A)^k$ for $k = 1, 2, \ldots$ we can determine the sizes of the Jordan blocks associated with λ

- factor out T and T^{-1}, $\lambda I - A = T(\lambda I - J)T^{-1}$

- for, say, a block of size 3:

$$\lambda_i I - J_i = \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \end{bmatrix} \quad (\lambda_i I - J_i)^2 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (\lambda_i I - J_i)^3 = 0$$

- for other blocks (say, size 3, for $k \geq 2$)

$$(\lambda_i I - J_j)^k = \begin{bmatrix} (\lambda_i - \lambda_j)^k & -k(\lambda_i - \lambda_j)^{k-1} & (k(k-1)/2)(\lambda_i - \lambda_j)^{k-2} \\ 0 & (\lambda_j - \lambda_i)^k & -k(\lambda_j - \lambda_i)^{k-1} \\ 0 & 0 & (\lambda_j - \lambda_i)^k \end{bmatrix}$$

Generalized eigenvectors

suppose $T^{-1}AT = J = \mathbf{diag}(J_1, \ldots, J_q)$

express T as

$$T = [T_1 \; T_2 \; \cdots \; T_q]$$

where $T_i \in \mathbf{C}^{n \times n_i}$ are the columns of T associated with ith Jordan block J_i

we have $AT_i = T_i J_i$

let $T_i = [v_{i1} \; v_{i2} \; \cdots \; v_{in_i}]$

then we have:

$$Av_{i1} = \lambda_i v_{i1},$$

$i.e.$, the first column of each T_i is an eigenvector associated with e.v. λ_i

for $j = 2, \ldots, n_i$,

$$Av_{ij} = v_{i\,j-1} + \lambda_i v_{ij}$$

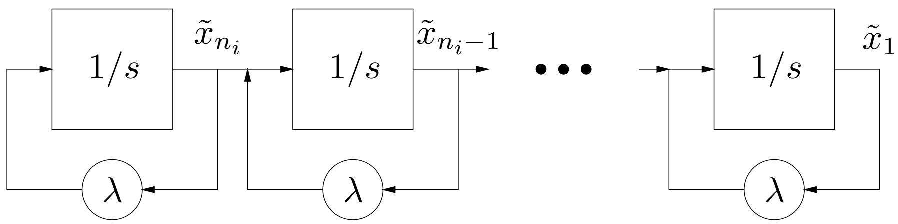
the vectors $v_{i1}, \ldots v_{in_i}$ are sometimes called *generalized eigenvectors*

Jordan form LDS

consider LDS $\dot{x} = Ax$

by change of coordinates $x = T\tilde{x}$, can put into form $\dot{\tilde{x}} = J\tilde{x}$

system is decomposed into independent 'Jordan block systems' $\dot{\tilde{x}}_i = J_i\tilde{x}_i$



Jordan blocks are sometimes called Jordan chains

(block diagram shows why)

Resolvent, exponential of Jordan block

resolvent of $k \times k$ Jordan block with eigenvalue λ:

$$\begin{aligned}
(sI - J_\lambda)^{-1} &= \begin{bmatrix} s-\lambda & -1 & & \\ & s-\lambda & \ddots & \\ & & \ddots & -1 \\ & & & s-\lambda \end{bmatrix}^{-1} \\
&= \begin{bmatrix} (s-\lambda)^{-1} & (s-\lambda)^{-2} & \cdots & (s-\lambda)^{-k} \\ & (s-\lambda)^{-1} & \cdots & (s-\lambda)^{-k+1} \\ & & \ddots & \vdots \\ & & & (s-\lambda)^{-1} \end{bmatrix} \\
&= (s-\lambda)^{-1}I + (s-\lambda)^{-2}F_1 + \cdots + (s-\lambda)^{-k}F_{k-1}
\end{aligned}$$

where F_i is the matrix with ones on the ith upper diagonal by inverse Laplace transform, exponential is:

$$
\begin{aligned}
e^{tJ_\lambda} &= e^{t\lambda}\left(I + tF_1 + \cdots + (t^{k-1}/(k-1)!)F_{k-1}\right) \\
&= e^{t\lambda}\begin{bmatrix} 1 & t & \cdots & t^{k-1}/(k-1)! \\ & 1 & \cdots & t^{k-2}/(k-2)! \\ & & \ddots & \vdots \\ & & & 1 \end{bmatrix}
\end{aligned}
$$

Jordan blocks yield:

- repeated poles in resolvent
- terms of form $t^p e^{t\lambda}$ in e^{tA}

Generalized modes

consider $\dot{x} = Ax$, with

$$x(0) = a_1 v_{i1} + \cdots + a_{n_i} v_{in_i} = T_i a$$

then $x(t) = Te^{Jt}\tilde{x}(0) = T_i e^{J_i t} a$

- trajectory stays in span of generalized eigenvectors
- coefficients have form $p(t)e^{\lambda t}$, where p is polynomial
- such solutions are called *generalized modes* of the system

with general $x(0)$ we can write

$$x(t) = e^{tA}x(0) = Te^{tJ}T^{-1}x(0) = \sum_{i=1}^{q} T_i e^{tJ_i}(S_i^T x(0))$$

where

$$T^{-1} = \begin{bmatrix} S_1^T \\ \vdots \\ S_q^T \end{bmatrix}$$

hence: all solutions of $\dot{x} = Ax$ are linear combinations of (generalized) modes

Cayley-Hamilton theorem

if $p(s) = a_0 + a_1 s + \cdots + a_k s^k$ is a polynomial and $A \in \mathbf{R}^{n\times n}$, we define

$$p(A) = a_0 I + a_1 A + \cdots + a_k A^k$$

Cayley-Hamilton theorem: for any $A \in \mathbf{R}^{n\times n}$ we have $\mathcal{X}(A) = 0$, where $\mathcal{X}(s) = \det(sI - A)$

example: with $A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ we have $\mathcal{X}(s) = s^2 - 5s - 2$, so

$$\begin{aligned} \mathcal{X}(A) &= A^2 - 5A - 2I \\ &= \begin{bmatrix} 7 & 10 \\ 15 & 22 \end{bmatrix} - 5\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} - 2I \\ &= 0 \end{aligned}$$

corollary: for every $p \in \mathbf{Z}_+$, we have

$$A^p \in \operatorname{span}\left\{ I,\ A,\ A^2, \ldots,\ A^{n-1} \right\}$$

(and if A is invertible, also for $p \in \mathbf{Z}$)

i.e., every power of A can be expressed as linear combination of $I,\ A, \ldots, A^{n-1}$

proof: divide $\mathcal{X}(s)$ into s^p to get $s^p = q(s)\mathcal{X}(s) + r(s)$

$r = \alpha_0 + \alpha_1 s + \cdots + \alpha_{n-1}s^{n-1}$ is remainder polynomial

then

$$A^p = q(A)\mathcal{X}(A) + r(A) = r(A) = \alpha_0 I + \alpha_1 A + \cdots + \alpha_{n-1}A^{n-1}$$

for $p = -1$: rewrite C-H theorem

$$\mathcal{X}(A) = A^n + a_{n-1}A^{n-1} + \cdots + a_0 I = 0$$

as

$$I = A\left(-(a_1/a_0)I - (a_2/a_0)A - \cdots - (1/a_0)A^{n-1}\right)$$

(A is invertible $\Leftrightarrow a_0 \neq 0$) so

$$A^{-1} = -(a_1/a_0)I - (a_2/a_0)A - \cdots - (1/a_0)A^{n-1}$$

i.e., inverse is linear combination of A^k, $k = 0, \ldots, n-1$

Proof of C-H theorem

first assume A is diagonalizable: $T^{-1}AT = \Lambda$

$$\mathcal{X}(s) = (s-\lambda_1)\cdots(s-\lambda_n)$$

since

$$\mathcal{X}(A) = \mathcal{X}(T\Lambda T^{-1}) = T\mathcal{X}(\Lambda)T^{-1}$$

it suffices to show $\mathcal{X}(\Lambda) = 0$

$$\begin{aligned}
\mathcal{X}(\Lambda) &= (\Lambda - \lambda_1 I)\cdots(\Lambda - \lambda_n I) \\
&= \mathbf{diag}(0, \lambda_2 - \lambda_1, \ldots, \lambda_n - \lambda_1)\cdots\mathbf{diag}(\lambda_1 - \lambda_n, \ldots, \lambda_{n-1} - \lambda_n, 0) \\
&= 0
\end{aligned}$$

now let's do general case: $T^{-1}AT = J$

$$\mathcal{X}(s) = (s-\lambda_1)^{n_1}\cdots(s-\lambda_q)^{n_q}$$

suffices to show $\mathcal{X}(J_i) = 0$

$$\mathcal{X}(J_i) = (J_i - \lambda_1 I)^{n_1}\cdots\underbrace{\begin{bmatrix} 0 & 1 & 0 & \cdots \\ 0 & 0 & 1 & \cdots \\ & & & \ddots \end{bmatrix}^{n_i}}_{(J_i - \lambda_i I)^{n_i}}\cdots(J_i - \lambda_q I)^{n_q} = 0$$

Lecture 13
Linear dynamical systems with inputs & outputs

- inputs & outputs: interpretations
- transfer matrix
- impulse and step matrices
- examples

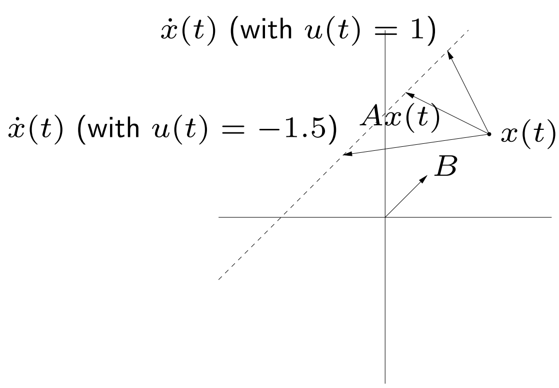
Inputs & outputs

recall continuous-time time-invariant LDS has form

$$\dot{x} = Ax + Bu, \qquad y = Cx + Du$$

- Ax is called the *drift term* (of $\dot{x}$)
- Bu is called the input term (of $\dot{x}$)

picture, with $B \in \mathbf{R}^{2\times 1}$:



Interpretations

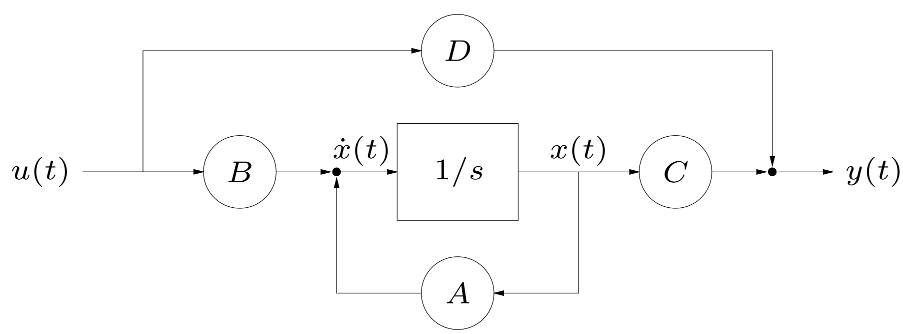
write $\dot{x} = Ax + b_1 u_1 + \cdots + b_m u_m$, where $B = [b_1 \ \cdots \ b_m]$

- state derivative is sum of autonomous term (Ax) and one term per input ($b_i u_i$)
- each input u_i gives another degree of freedom for $\dot{x}$ (assuming columns of B independent)

write $\dot{x} = Ax + Bu$ as $\dot{x}_i = \tilde{a}_i^T x + \tilde{b}_i^T u$, where $\tilde{a}_i^T$, $\tilde{b}_i^T$ are the rows of A, B

- ith state derivative is linear function of state x and input u

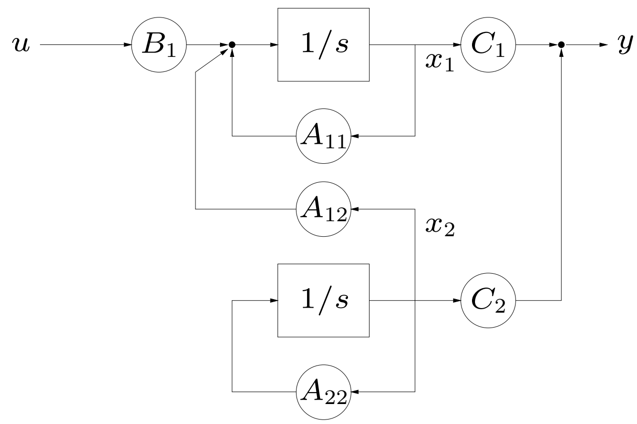
Block diagram



- A_{ij} is gain factor from state x_j into integrator i
- B_{ij} is gain factor from input u_j into integrator i
- C_{ij} is gain factor from state x_j into output y_i
- D_{ij} is gain factor from input u_j into output y_i

interesting when there is structure, *e.g.*, with $x_1 \in \mathbf{R}^{n_1}$, $x_2 \in \mathbf{R}^{n_2}$:

$$\frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} B_1 \\ 0 \end{bmatrix} u, \qquad y = \begin{bmatrix} C_1 & C_2 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$



- x_2 is not affected by input u, *i.e.*, x_2 propagates autonomously
- x_2 affects y directly and through x_1

Transfer matrix

take Laplace transform of $\dot{x} = Ax + Bu$:

$$sX(s) - x(0) = AX(s) + BU(s)$$

hence

$$X(s) = (sI - A)^{-1}x(0) + (sI - A)^{-1}BU(s)$$

so

$$x(t) = e^{tA}x(0) + \int_0^t e^{(t-\tau)A}Bu(\tau)\, d\tau$$

- $e^{tA}x(0)$ is the unforced or autonomous response
- $e^{tA}B$ is called the input-to-state impulse matrix
- $(sI - A)^{-1}B$ is called the *input-to-state transfer matrix* or *transfer function*

with $y = Cx + Du$ we have:

$$Y(s) = C(sI - A)^{-1}x(0) + (C(sI - A)^{-1}B + D)U(s)$$

so

$$y(t) = Ce^{tA}x(0) + \int_0^t Ce^{(t-\tau)A}Bu(\tau)\, d\tau + Du(t)$$

- output term $Ce^{tA}x(0)$ due to initial condition
- $H(s) = C(sI - A)^{-1}B + D$ is called the *transfer function* or *transfer matrix*
- $h(t) = Ce^{tA}B + D\delta(t)$ is called the *impulse matrix* or *impulse response* (δ is the Dirac delta function)

with zero initial condition we have:

$$Y(s) = H(s)U(s), \qquad y = h * u$$

where $*$ is convolution (of matrix valued functions)

intepretation:

- H_{ij} is transfer function from input u_j to output y_i

Impulse matrix

impulse matrix $h(t) = Ce^{tA}B + D\delta(t)$

with $x(0) = 0$, $y = h * u$, *i.e.*,

$$y_i(t) = \sum_{j=1}^{m} \int_0^t h_{ij}(t-\tau)u_j(\tau)\, d\tau$$

interpretations:

- $h_{ij}(t)$ is impulse response from jth input to ith output
- $h_{ij}(t)$ gives y_i when $u(t) = e_j\delta$
- $h_{ij}(\tau)$ shows how dependent output i is, on what input j was, τ seconds ago
- i indexes output; j indexes input; τ indexes time lag

Step matrix

the *step matrix* or *step response matrix* is given by

$$s(t) = \int_0^t h(\tau)\, d\tau$$

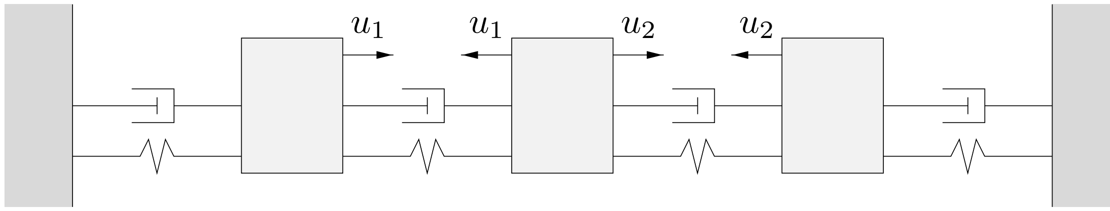
interpretations:

- $s_{ij}(t)$ is step response from jth input to ith output
- $s_{ij}(t)$ gives y_i when $u = e_j$ for $t \geq 0$

for invertible A, we have

$$s(t) = CA^{-1}\left(e^{tA} - I\right)B + D$$

Example 1



- unit masses, springs, dampers
- u_1 is tension between 1st & 2nd masses
- u_2 is tension between 2nd & 3rd masses
- $y \in \mathbf{R}^3$ is displacement of masses 1,2,3
- $x = \begin{bmatrix} y \\ \dot{y} \end{bmatrix}$

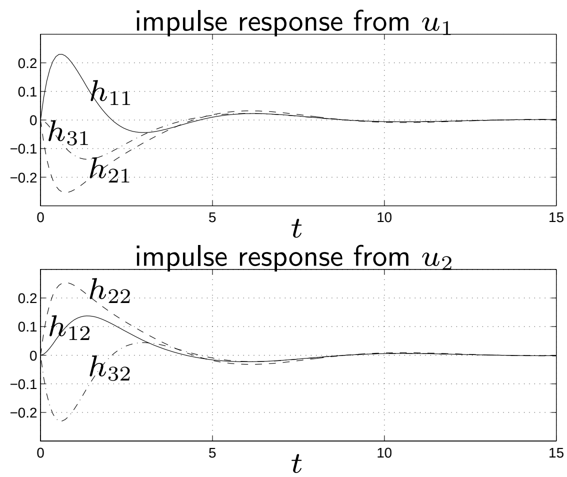
system is:

$$\dot{x} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -2 & 1 & 0 & -2 & 1 & 0 \\ 1 & -2 & 1 & 1 & -2 & 1 \\ 0 & 1 & -2 & 0 & 1 & -2 \end{bmatrix} x + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ -1 & 1 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

eigenvalues of A are

$$-1.71 \pm j0.71, \qquad -1.00 \pm j1.00, \qquad -0.29 \pm j0.71$$

impulse matrix:

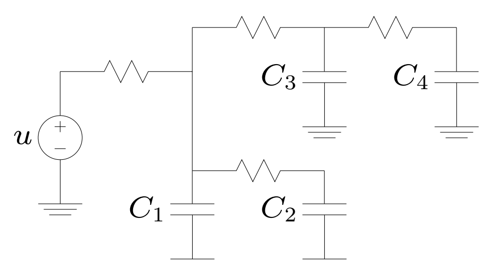


roughly speaking:

- impulse at u_1 affects third mass less than other two
- impulse at u_2 affects first mass later than other two

Example 2

interconnect circuit:



- $u(t) \in \mathbf{R}$ is input (drive) voltage
- x_i is voltage across C_i
- output is state: $y = x$
- unit resistors, unit capacitors
- step response matrix shows delay to each node

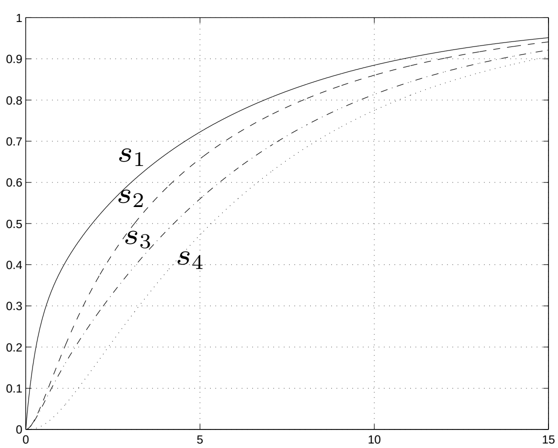
system is

$$\dot{x} = \begin{bmatrix} -3 & 1 & 1 & 0 \\ 1 & -1 & 0 & 0 \\ 1 & 0 & -2 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix} x + \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} u, \qquad y = x$$

eigenvalues of A are

$$-0.17, \qquad -0.66, \qquad -2.21, \qquad -3.96$$

step response matrix $s(t) \in \mathbf{R}^{4\times 1}$:



- shortest delay to x_1; longest delay to x_4
- delays ≈ 10, consistent with slowest ($i.e.$, dominant) eigenvalue -0.17

DC or static gain matrix

- transfer matrix at $s = 0$ is $H(0) = -CA^{-1}B + D \in \mathbf{R}^{m \times p}$
- DC transfer matrix describes system under *static* conditions, *i.e.*, x, u, y constant:
$$0 = \dot{x} = Ax + Bu, \qquad y = Cx + Du$$
eliminate x to get $y = H(0)u$
- if system is stable,
$$H(0) = \int_0^\infty h(t)\, dt = \lim_{t \to \infty} s(t)$$
(recall: $H(s) = \int_0^\infty e^{-st} h(t)\, dt$, $s(t) = \int_0^t h(\tau)\, d\tau$)
if $u(t) \to u_\infty \in \mathbf{R}^m$, then $y(t) \to y_\infty \in \mathbf{R}^p$ where $y_\infty = H(0)u_\infty$

DC gain matrix for example 1 (springs):

$$H(0) = \begin{bmatrix} 1/4 & 1/4 \\ -1/2 & 1/2 \\ -1/4 & -1/4 \end{bmatrix}$$

DC gain matrix for example 2 (RC circuit):

$$H(0) = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

(do these make sense?)

Discretization with piecewise constant inputs

linear system $\dot{x} = Ax + Bu$, $y = Cx + Du$

suppose $u_d : \mathbf{Z}_+ \to \mathbf{R}^m$ is a sequence, and

$$u(t) = u_d(k) \qquad \text{for} \;\; kh \le t < (k+1)h, \quad k = 0, 1, \ldots$$

define sequences

$$x_d(k) = x(kh), \qquad y_d(k) = y(kh), \qquad k = 0, 1, \ldots$$

- $h > 0$ is called the *sample interval* (for x and y) or *update interval* (for u)
- u is piecewise constant (called *zero-order-hold*)
- x_d, y_d are sampled versions of x, y

$$\begin{aligned} x_d(k+1) &= x((k+1)h) \\ &= e^{hA}x(kh) + \int_0^h e^{\tau A} B u((k+1)h - \tau)\, d\tau \\ &= e^{hA}x_d(k) + \left(\int_0^h e^{\tau A}\, d\tau \right) B\, u_d(k) \end{aligned}$$

x_d, u_d, and y_d satisfy discrete-time LDS equations

$$x_d(k+1) = A_d x_d(k) + B_d u_d(k), \qquad y_d(k) = C_d x_d(k) + D_d u_d(k)$$

where

$$A_d = e^{hA}, \qquad B_d = \left(\int_0^h e^{\tau A}\, d\tau \right) B, \qquad C_d = C, \qquad D_d = D$$

called *discretized system*

if A is invertible, we can express integral as

$$\int_0^h e^{\tau A}\, d\tau = A^{-1}\left(e^{hA} - I\right)$$

stability: if eigenvalues of A are $\lambda_1, \ldots, \lambda_n$, then eigenvalues of A_d are $e^{h\lambda_1}, \ldots, e^{h\lambda_n}$

discretization preserves stability properties since

$$\Re\lambda_i < 0 \quad \Leftrightarrow \quad \left|e^{h\lambda_i}\right| < 1$$

for $h > 0$

extensions/variations:

- *offsets:* updates for u and sampling of x, y are offset in time
- *multirate:* u_i updated, y_i sampled at different intervals
  (usually integer multiples of a common interval h)

both very common in practice

Dual system

the *dual system* associated with system

$$\dot{x} = Ax + Bu, \qquad y = Cx + Du$$

is given by

$$\dot{z} = A^T z + C^T v, \qquad w = B^T z + D^T v$$

- all matrices are transposed
- role of B and C are swapped

transfer function of dual system:

$$(B^T)(sI - A^T)^{-1}(C^T) + D^T = H(s)^T$$

where $H(s) = C(sI - A)^{-1}B + D$

(for SISO case, TF of dual is same as original)

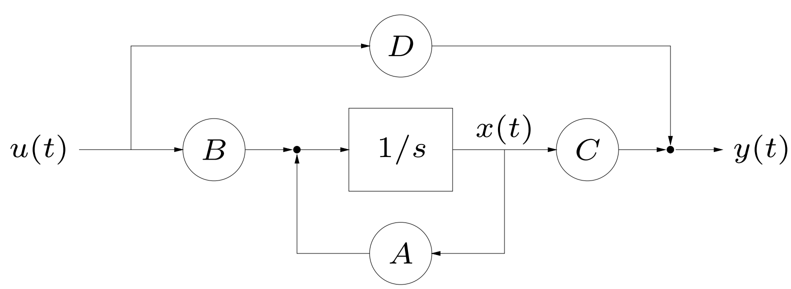
eigenvalues (hence stability properties) are the same

Dual via block diagram

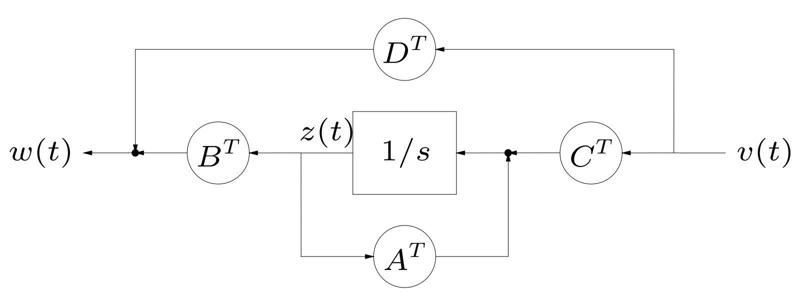
in terms of block diagrams, dual is formed by:

- transpose all matrices
- swap inputs and outputs on all boxes
- reverse directions of signal flow arrows
- swap solder joints and summing junctions

original system:



dual system:



Causality

interpretation of

$$\begin{aligned} x(t) &= e^{tA}x(0) + \int_0^t e^{(t-\tau)A}Bu(\tau)\, d\tau \\ y(t) &= Ce^{tA}x(0) + \int_0^t Ce^{(t-\tau)A}Bu(\tau)\, d\tau + Du(t) \end{aligned}$$

for $t \geq 0$:

current state ($x(t)$) and output ($y(t)$) depend on *past* input ($u(\tau)$ for $\tau \leq t$)

$i.e.$, mapping from input to state and output is *causal* (with fixed *initial* state)

now consider fixed *final* state $x(T)$: for $t \leq T$,

$$x(t) = e^{(t-T)A}x(T) + \int_T^t e^{(t-\tau)A}Bu(\tau)\, d\tau,$$

$i.e.$, current state (and output) depend on future input!

so for fixed final condition, same system is anti-causal

Idea of state

$x(t)$ is called *state* of system at time t since:

- future output depends only on current state and future input
- future output depends on past input only through current state
- state summarizes effect of past inputs on future output
- state is bridge between past inputs and future outputs

Change of coordinates

start with LDS $\dot{x} = Ax + Bu$, $y = Cx + Du$

change coordinates in $\mathbf{R}^n$ to $\tilde{x}$, with $x = T\tilde{x}$

then

$$\dot{\tilde{x}} = T^{-1}\dot{x} = T^{-1}(Ax + Bu) = T^{-1}AT\tilde{x} + T^{-1}Bu$$

hence LDS can be expressed as

$$\dot{\tilde{x}} = \tilde{A}\tilde{x} + \tilde{B}u, \qquad y = \tilde{C}\tilde{x} + \tilde{D}u$$

where

$$\tilde{A} = T^{-1}AT, \qquad \tilde{B} = T^{-1}B, \qquad \tilde{C} = CT, \qquad \tilde{D} = D$$

TF is same (since u, y aren't affected):

$$\tilde{C}(sI - \tilde{A})^{-1}\tilde{B} + \tilde{D} = C(sI - A)^{-1}B + D$$

Standard forms for LDS

can change coordinates to put A in various forms (diagonal, real modal, Jordan . . .)

e.g., to put LDS in *diagonal form*, find T s.t.

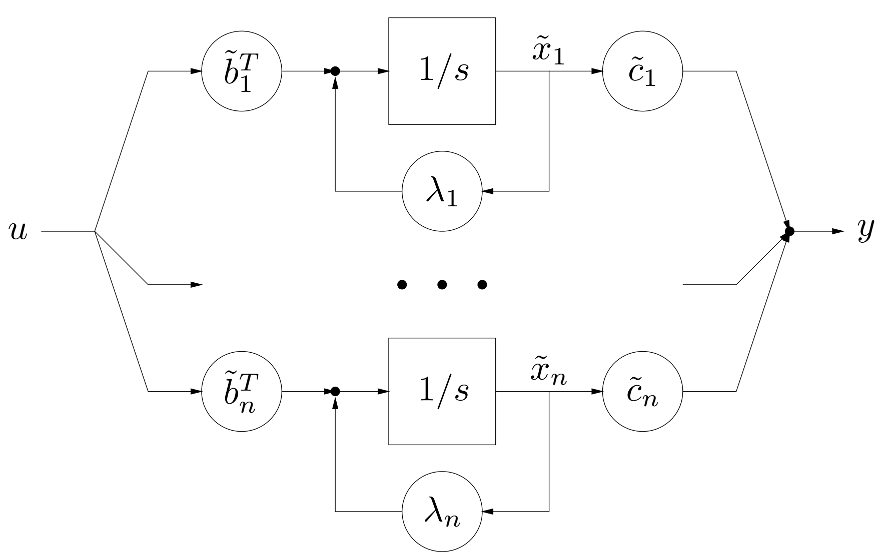
$$T^{-1}AT = \mathbf{diag}(\lambda_1, \ldots, \lambda_n)$$

write

$$T^{-1}B = \begin{bmatrix} \tilde{b}_1^T \\ \vdots \\ \tilde{b}_n^T \end{bmatrix}, \qquad CT = \begin{bmatrix} \tilde{c}_1 & \cdots & \tilde{c}_n \end{bmatrix}$$

so

$$\dot{\tilde{x}}_i = \lambda_i \tilde{x}_i + \tilde{b}_i^T u, \qquad y = \sum_{i=1}^n \tilde{c}_i \tilde{x}_i$$

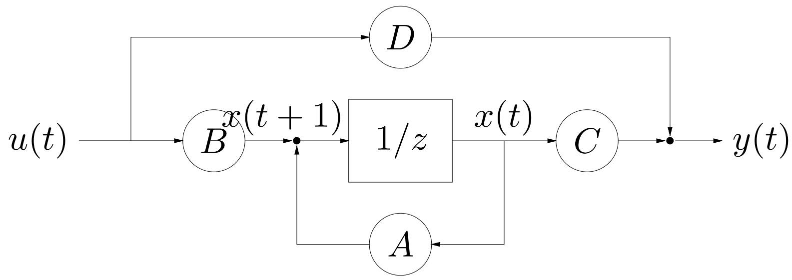


(here we assume $D = 0$)

Discrete-time systems

discrete-time LDS:

$$x(t+1) = Ax(t) + Bu(t), \qquad y(t) = Cx(t) + Du(t)$$



- only difference w/cts-time: z instead of s
- interpretation of z^{-1} block:
  - unit delayor (shifts sequence back in time one epoch)
  - latch (plus small delay to avoid race condition)

we have:

$$x(1) = Ax(0) + Bu(0),$$

$$\begin{aligned} x(2) &= Ax(1) + Bu(1) \\ &= A^2x(0) + ABu(0) + Bu(1), \end{aligned}$$

and in general, for $t \in \mathbf{Z}_+$,

$$x(t) = A^t x(0) + \sum_{\tau=0}^{t-1} A^{(t-1-\tau)} Bu(\tau)$$

hence

$$y(t) = CA^t x(0) + h * u$$

where $*$ is discrete-time convolution and

$$h(t) = \begin{cases} D, & t = 0 \\ CA^{t-1}B, & t > 0 \end{cases}$$

is the impulse response

$\mathcal{Z}$-transform

suppose $w \in \mathbf{R}^{p\times q}$ is a sequence (discrete-time signal), *i.e.*,

$$w : \mathbf{Z}_+ \to \mathbf{R}^{p\times q}$$

recall $\mathcal{Z}$-transform $W = \mathcal{Z}(w)$:

$$W(z) = \sum_{t=0}^{\infty} z^{-t} w(t)$$

where $W : D \subseteq \mathbf{C} \to \mathbf{C}^{p\times q}$ (D is domain of W)

time-advanced or shifted signal v:

$$v(t) = w(t+1), \qquad t = 0, 1, \ldots$$

$\mathcal{Z}$-transform of time-advanced signal:

$$\begin{aligned} V(z) &= \sum_{t=0}^{\infty} z^{-t} w(t+1) \\ &= z \sum_{t=1}^{\infty} z^{-t} w(t) \\ &= zW(z) - zw(0) \end{aligned}$$

Discrete-time transfer function

take $\mathcal{Z}$-transform of system equations

$$x(t+1) = Ax(t) + Bu(t), \qquad y(t) = Cx(t) + Du(t)$$

yields

$$zX(z) - zx(0) = AX(z) + BU(z), \qquad Y(z) = CX(z) + DU(z)$$

solve for $X(z)$ to get

$$X(z) = (zI - A)^{-1} zx(0) + (zI - A)^{-1} BU(z)$$

(note extra z in first term!)

hence

$$Y(z) = H(z)U(z) + C(zI - A)^{-1}zx(0)$$

where $H(z) = C(zI - A)^{-1}B + D$ is the *discrete-time transfer function*

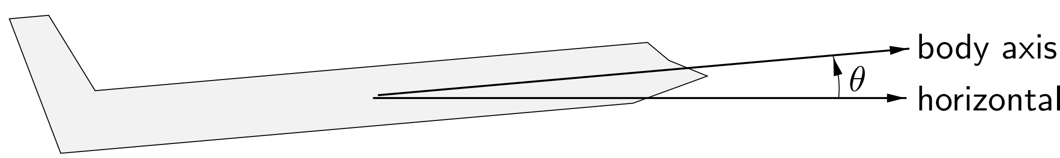
note power series expansion of resolvent:

$$(zI - A)^{-1} = z^{-1}I + z^{-2}A + z^{-3}A^2 + \cdots$$

Lecture 14
Example: Aircraft dynamics

- longitudinal aircraft dynamics
- wind gust & control inputs
- linearized dynamics
- steady-state analysis
- eigenvalues & modes
- impulse matrices

Longitudinal aircraft dynamics



variables are (small) deviations from operating point or *trim conditions*

state (components):

- u: velocity of aircraft along body axis
- v: velocity of aircraft perpendicular to body axis
  (down is positive)
- θ: angle between body axis and horizontal
  (up is positive)
- $q = \dot{\theta}$: angular velocity of aircraft (pitch rate)

Inputs

disturbance inputs:

- u_w: velocity of wind along body axis
- v_w: velocity of wind perpendicular to body axis

control or actuator inputs:

- δ_e: elevator angle ($\delta_e > 0$ is down)
- δ_t: thrust

Linearized dynamics

for 747, level flight, 40000 ft, 774 ft/sec,

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{q} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} -.003 & .039 & 0 & -.322 \\ -.065 & -.319 & 7.74 & 0 \\ .020 & -.101 & -.429 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} u - u_w \\ v - v_w \\ q \\ \theta \end{bmatrix} + \begin{bmatrix} .01 & 1 \\ -.18 & -.04 \\ -1.16 & .598 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_e \\ \delta_t \end{bmatrix}$$

- units: ft, sec, crad ($= 0.01$rad $\approx 0.57°$)
- matrix coefficients are called *stability derivatives*

outputs of interest:

- aircraft speed u (deviation from trim)
- climb rate $\dot{h} = -v + 7.74\theta$

Steady-state analysis

DC gain from $(u_w, v_w, \delta_e, \delta_t)$ to $(u, \dot{h})$:

$$H(0) = -CA^{-1}B + D = \begin{bmatrix} 1 & 0 & 27.2 & -15.0 \\ 0 & -1 & -1.34 & 24.9 \end{bmatrix}$$

gives steady-state change in speed & climb rate due to wind, elevator & thrust changes

solve for control variables in terms of wind velocities, desired speed & climb rate

$$\begin{bmatrix} \delta_e \\ \delta_t \end{bmatrix} = \begin{bmatrix} .0379 & .0229 \\ .0020 & .0413 \end{bmatrix} \begin{bmatrix} u - u_w \\ \dot{h} + v_w \end{bmatrix}$$

- level flight, increase in speed is obtained mostly by increasing elevator ($i.e.$, downwards)

- constant speed, increase in climb rate is obtained by increasing thrust and increasing elevator ($i.e.$, downwards)

(thrust on 747 gives strong pitch up torque)

Eigenvalues and modes

eigenvalues are

$$-0.3750 \pm 0.8818j, \quad -0.0005 \pm 0.0674j$$

- two complex modes, called *short-period* and *phugoid*, respectively

- system is stable (but lightly damped)

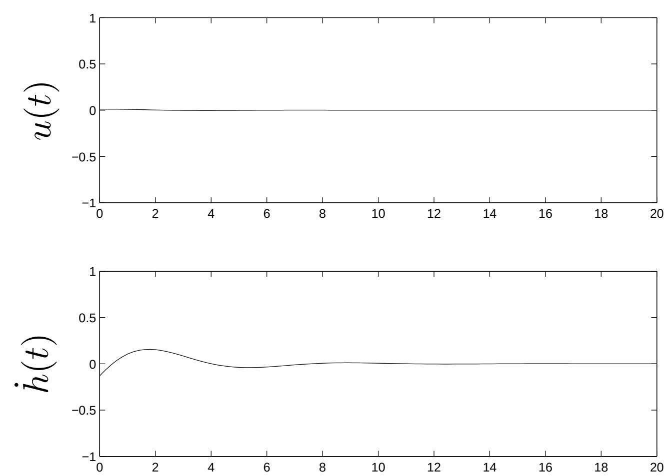
- hence step responses converge (eventually) to DC gain matrix eigenvectors are

$$x_{\text{short}} = \begin{bmatrix} 0.0005 \\ -0.5433 \\ -0.0899 \\ -0.0283 \end{bmatrix} \pm j \begin{bmatrix} 0.0135 \\ 0.8235 \\ -0.0677 \\ 0.1140 \end{bmatrix},$$

$$x_{\text{phug}} = \begin{bmatrix} -0.7510 \\ -0.0962 \\ -0.0111 \\ 0.1225 \end{bmatrix} \pm j \begin{bmatrix} 0.6130 \\ 0.0941 \\ 0.0082 \\ 0.1637 \end{bmatrix}$$

Short-period mode

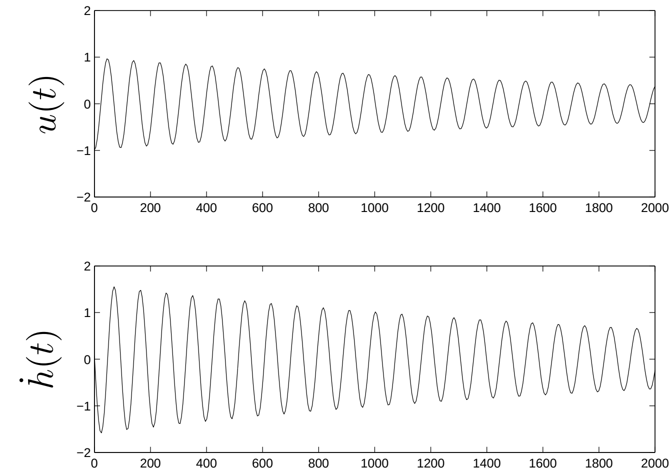
$y(t) = Ce^{tA}(\Re x_{\text{short}})$ (pure short-period mode motion)



- only small effect on speed u
- period ≈ 7 sec, decays in ≈ 10 sec

Phugoid mode

$y(t) = Ce^{tA}(\Re x_{\text{phug}})$ (pure phugoid mode motion)

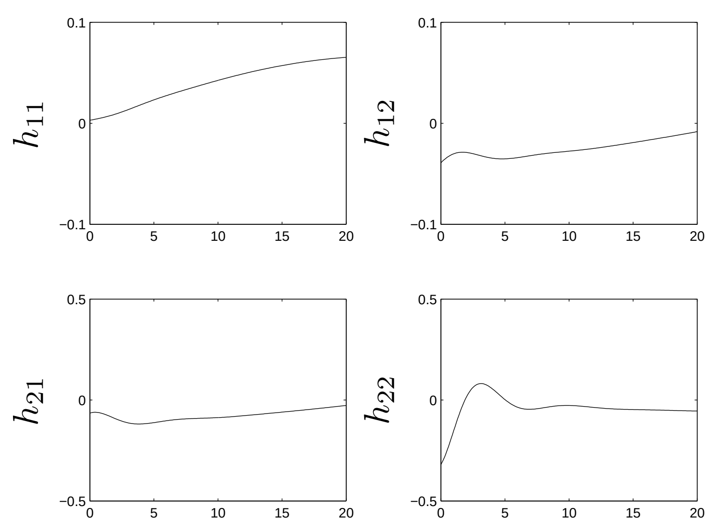


- affects both speed and climb rate
- period ≈ 100 sec; decays in ≈ 5000 sec

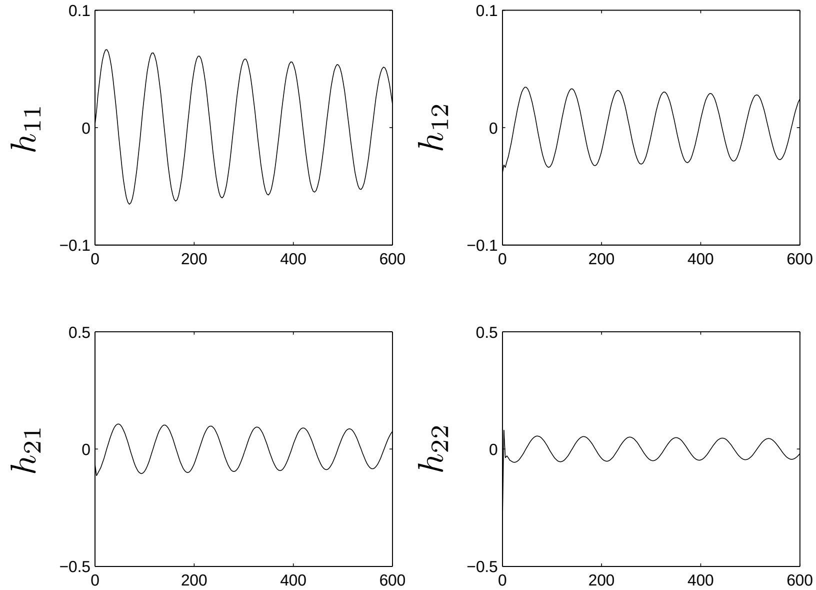
Dynamic response to wind gusts

impulse response matrix from (u_w, v_w) to $(u, \dot{h})$ (gives response to short wind bursts)

over time period $[0, 20]$:



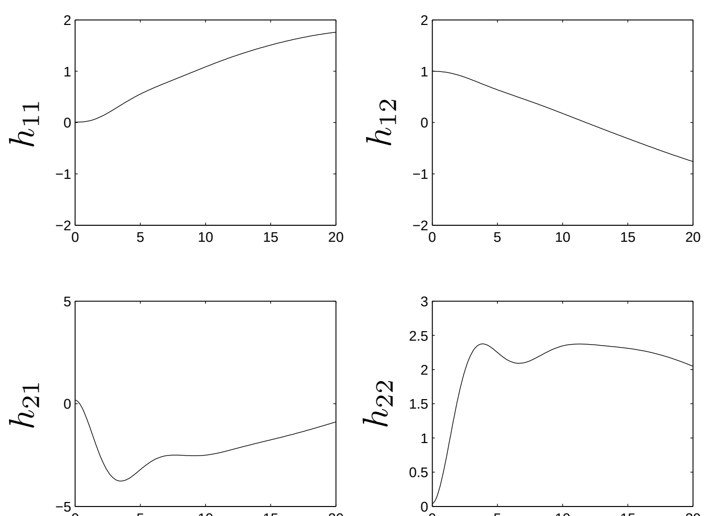
over time period $[0, 600]$:



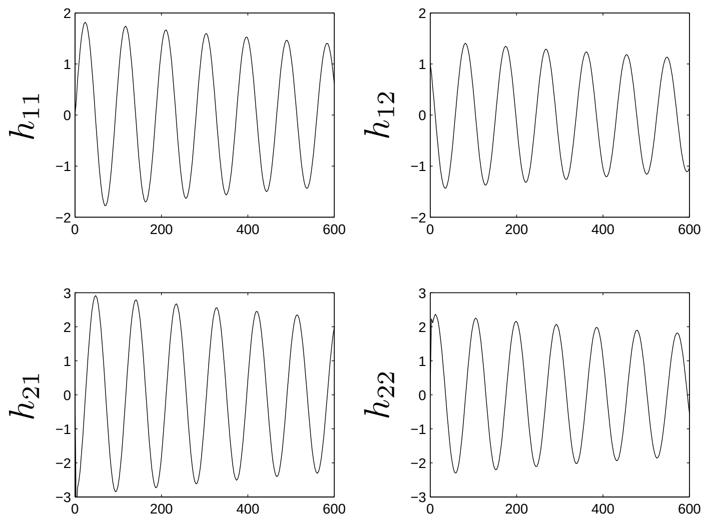
Dynamic response to actuators

impulse response matrix from (δ_e, δ_t) to $(u, \dot{h})$

over time period $[0, 20]$:



over time period $[0, 600]$:



Lecture 15
Symmetric matrices, quadratic forms, matrix norm, and SVD

- eigenvectors of symmetric matrices
- quadratic forms
- inequalities for quadratic forms
- positive semidefinite matrices
- norm of a matrix
- singular value decomposition

Eigenvalues of symmetric matrices

suppose $A \in \mathbf{R}^{n \times n}$ is symmetric, $i.e.$, $A = A^T$

fact: the eigenvalues of A are real

to see this, suppose $Av = \lambda v$, $v \neq 0$, $v \in \mathbf{C}^n$

then

$$\overline{v}^T A v = \overline{v}^T (Av) = \lambda \overline{v}^T v = \lambda \sum_{i=1}^{n} |v_i|^2$$

but also

$$\overline{v}^T A v = \overline{(Av)}^T v = \overline{(\lambda v)}^T v = \overline{\lambda} \sum_{i=1}^{n} |v_i|^2$$

so we have $\lambda = \overline{\lambda}$, $i.e.$, $\lambda \in \mathbf{R}$ (hence, can assume $v \in \mathbf{R}^n$)

Eigenvectors of symmetric matrices

fact: there is a set of orthonormal eigenvectors of A, *i.e.*, $q_1, \ldots, q_n$ s.t. $Aq_i = \lambda_i q_i$, $q_i^T q_j = \delta_{ij}$

in matrix form: there is an orthogonal Q s.t.

$$Q^{-1}AQ = Q^TAQ = \Lambda$$

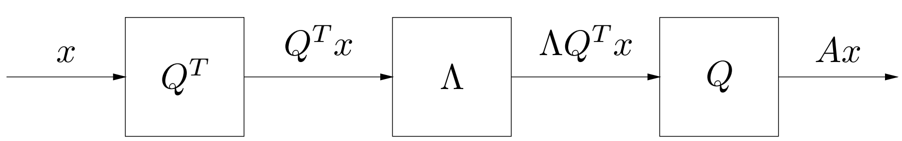
hence we can express A as

$$A = Q\Lambda Q^T = \sum_{i=1}^{n} \lambda_i q_i q_i^T$$

in particular, q_i are both left and right eigenvectors

Interpretations

$A = Q\Lambda Q^T$



linear mapping $y = Ax$ can be decomposed as

- resolve into q_i coordinates
- scale coordinates by λ_i
- reconstitute with basis q_i

or, geometrically,

- rotate by Q^T
- diagonal real scale ('dilation') by Λ
- rotate back by Q

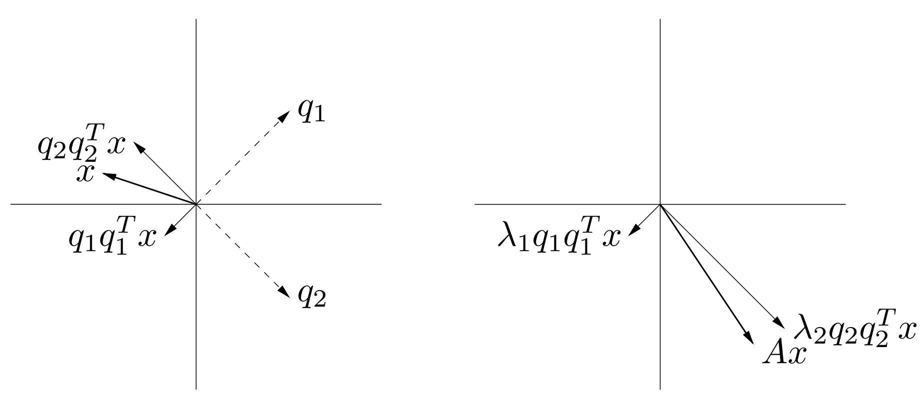
decomposition

$$A = \sum_{i=1}^{n} \lambda_i q_i q_i^T$$

expresses A as linear combination of 1-dimensional projections

example:

$$\begin{aligned} A &= \begin{bmatrix} -1/2 & 3/2 \\ 3/2 & -1/2 \end{bmatrix} \\ &= \left(\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\right)\begin{bmatrix} 1 & 0 \\ 0 & -2 \end{bmatrix}\left(\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\right)^T \end{aligned}$$



proof (case of λ_i distinct)

since λ_i distinct, can find $v_1, \ldots, v_n$, a set of linearly independent eigenvectors of A:

$$Av_i = \lambda_i v_i, \qquad \|v_i\| = 1$$

then we have

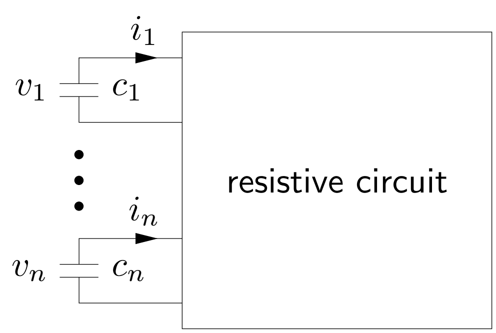
$$v_i^T(Av_j) = \lambda_j v_i^T v_j = (Av_i)^T v_j = \lambda_i v_i^T v_j$$

so $(\lambda_i - \lambda_j)v_i^T v_j = 0$

for $i \neq j$, $\lambda_i \neq \lambda_j$, hence $v_i^T v_j = 0$

- in this case we can say: eigenvectors *are* orthogonal
- in general case (λ_i not distinct) we must say: eigenvectors *can be chosen* to be orthogonal

Example: RC circuit



$$c_k \dot{v}_k = -i_k, \qquad i = Gv$$

$G = G^T \in \mathbf{R}^{n\times n}$ is conductance matrix of resistive circuit

thus $\dot{v} = -C^{-1}Gv$ where $C = \mathbf{diag}(c_1, \ldots, c_n)$

note $-C^{-1}G$ is not symmetric

use state $x_i = \sqrt{c_i} v_i$, so

$$\dot{x} = C^{1/2}\dot{v} = -C^{-1/2}GC^{-1/2}x$$

where $C^{1/2} = \mathbf{diag}(\sqrt{c_1}, \ldots, \sqrt{c_n})$

we conclude:

- eigenvalues $\lambda_1, \ldots, \lambda_n$ of $-C^{-1/2}GC^{-1/2}$ (hence, $-C^{-1}G$) are real
- eigenvectors q_i (in x_i coordinates) can be chosen orthogonal
- eigenvectors in voltage coordinates, $s_i = C^{-1/2}q_i$, satisfy

$$-C^{-1}Gs_i = \lambda_i s_i, \qquad s_i^T C s_i = \delta_{ij}$$

Quadratic forms

a function $f : \mathbf{R}^n \to \mathbf{R}$ of the form

$$f(x) = x^T A x = \sum_{i,j=1}^{n} A_{ij} x_i x_j$$

is called a *quadratic form*

in a quadratic form we may as well assume $A = A^T$ since

$$x^T A x = x^T((A + A^T)/2)x$$

($(A + A^T)/2$ is called the *symmetric part* of A)

uniqueness: if $x^T A x = x^T B x$ for all $x \in \mathbf{R}^n$ and $A = A^T$, $B = B^T$, then $A = B$

Examples

- $\|Bx\|^2 = x^T B^T Bx$
- $\sum_{i=1}^{n-1}(x_{i+1} - x_i)^2$
- $\|Fx\|^2 - \|Gx\|^2$

sets defined by quadratic forms:

- $\{ x \mid f(x) = a \}$ is called a *quadratic surface*
- $\{ x \mid f(x) \leq a \}$ is called a *quadratic region*

Inequalities for quadratic forms

suppose $A = A^T$, $A = Q\Lambda Q^T$ with eigenvalues sorted so $\lambda_1 \geq \cdots \geq \lambda_n$

$$\begin{aligned} x^T Ax &= x^T Q\Lambda Q^T x \\ &= (Q^T x)^T \Lambda (Q^T x) \\ &= \sum_{i=1}^{n} \lambda_i (q_i^T x)^2 \\ &\leq \lambda_1 \sum_{i=1}^{n} (q_i^T x)^2 \\ &= \lambda_1 \|x\|^2 \end{aligned}$$

$i.e.$, we have $x^T Ax \leq \lambda_1 x^T x$

similar argument shows $x^TAx \geq \lambda_n\|x\|^2$, so we have

$$\lambda_n x^T x \leq x^T A x \leq \lambda_1 x^T x$$

sometimes λ_1 is called $\lambda_{\max}$, λ_n is called $\lambda_{\min}$

note also that

$$q_1^T A q_1 = \lambda_1 \|q_1\|^2, \qquad q_n^T A q_n = \lambda_n \|q_n\|^2,$$

so the inequalities are tight

Positive semidefinite and positive definite matrices

suppose $A = A^T \in \mathbf{R}^{n\times n}$

we say A is *positive semidefinite* if $x^TAx \geq 0$ for all x

- denoted $A \geq 0$ (and sometimes $A \succeq 0$)
- $A \geq 0$ if and only if $\lambda_{\min}(A) \geq 0$, *i.e.*, all eigenvalues are nonnegative
- **not** the same as $A_{ij} \geq 0$ for all i, j

we say A is *positive definite* if $x^TAx > 0$ for all $x \neq 0$

- denoted $A > 0$
- $A > 0$ if and only if $\lambda_{\min}(A) > 0$, *i.e.*, all eigenvalues are positive

Matrix inequalities

- we say A is *negative semidefinite* if $-A \geq 0$
- we say A is *negative definite* if $-A > 0$
- otherwise, we say A is *indefinite*

matrix inequality: if $B = B^T \in \mathbf{R}^n$ we say $A \geq B$ if $A - B \geq 0$, $A < B$ if $B - A > 0$, etc.

for example:

- $A \geq 0$ means A is positive semidefinite
- $A > B$ means $x^T A x > x^T B x$ for all $x \neq 0$

many properties that you'd guess hold actually do, $e.g.$,

- if $A \geq B$ and $C \geq D$, then $A + C \geq B + D$
- if $B \leq 0$ then $A + B \leq A$
- if $A \geq 0$ and $\alpha \geq 0$, then $\alpha A \geq 0$
- $A^2 \geq 0$
- if $A > 0$, then $A^{-1} > 0$

matrix inequality is only a *partial order*: we can have

$$A \not\geq B, \qquad B \not\geq A$$

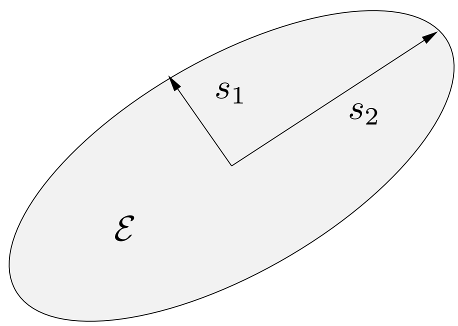
(such matrices are called *incomparable*)

Ellipsoids

if $A = A^T > 0$, the set

$$\mathcal{E} = \{ x \mid x^T A x \leq 1 \}$$

is an *ellipsoid* in $\mathbf{R}^n$, centered at 0



semi-axes are given by $s_i = \lambda_i^{-1/2} q_i$, *i.e.*:

- eigenvectors determine directions of semiaxes
- eigenvalues determine lengths of semiaxes

note:

- in direction q_1, $x^T A x$ is *large*, hence ellipsoid is *thin* in direction q_1
- in direction q_n, $x^T A x$ is *small*, hence ellipsoid is *fat* in direction q_n
- $\sqrt{\lambda_{\max}/\lambda_{\min}}$ gives maximum *eccentricity*

if $\tilde{\mathcal{E}} = \{ x \mid x^T B x \leq 1 \}$, where $B > 0$, then $\mathcal{E} \subseteq \tilde{\mathcal{E}} \iff A \geq B$

Gain of a matrix in a direction

suppose $A \in \mathbf{R}^{m\times n}$ (not necessarily square or symmetric)

for $x \in \mathbf{R}^n$, $\|Ax\|/\|x\|$ gives the *amplification factor* or *gain* of A in the direction x

obviously, gain varies with direction of input x

questions:

- what is maximum gain of A
  (and corresponding maximum gain direction)?
- what is minimum gain of A
  (and corresponding minimum gain direction)?
- how does gain of A vary with direction?

Matrix norm

the maximum gain

$$\max_{x\neq 0} \frac{\|Ax\|}{\|x\|}$$

is called the *matrix norm* or *spectral norm* of A and is denoted $\|A\|$

$$\max_{x\neq 0} \frac{\|Ax\|^2}{\|x\|^2} = \max_{x\neq 0} \frac{x^TA^TAx}{\|x\|^2} = \lambda_{\max}(A^TA)$$

so we have $\|A\| = \sqrt{\lambda_{\max}(A^TA)}$

similarly the minimum gain is given by

$$\min_{x\neq 0} \|Ax\|/\|x\| = \sqrt{\lambda_{\min}(A^TA)}$$

note that

- $A^TA \in \mathbf{R}^{n\times n}$ is symmetric and $A^TA \geq 0$ so $\lambda_{\min},\ \lambda_{\max} \geq 0$
- 'max gain' input direction is $x = q_1$, eigenvector of A^TA associated with $\lambda_{\max}$
- 'min gain' input direction is $x = q_n$, eigenvector of A^TA associated with $\lambda_{\min}$

example: $A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$

$$\begin{aligned} A^TA &= \begin{bmatrix} 35 & 44 \\ 44 & 56 \end{bmatrix} \\ &= \begin{bmatrix} 0.620 & 0.785 \\ 0.785 & -0.620 \end{bmatrix} \begin{bmatrix} 90.7 & 0 \\ 0 & 0.265 \end{bmatrix} \begin{bmatrix} 0.620 & 0.785 \\ 0.785 & -0.620 \end{bmatrix}^T \end{aligned}$$

then $\|A\| = \sqrt{\lambda_{\max}(A^TA)} = 9.53$:

$$\left\| \begin{bmatrix} 0.620 \\ 0.785 \end{bmatrix} \right\| = 1, \qquad \left\| A \begin{bmatrix} 0.620 \\ 0.785 \end{bmatrix} \right\| = \left\| \begin{bmatrix} 2.18 \\ 4.99 \\ 7.78 \end{bmatrix} \right\| = 9.53$$

min gain is $\sqrt{\lambda_{\min}(A^TA)} = 0.514$:

$$\left\| \begin{bmatrix} 0.785 \\ -0.620 \end{bmatrix} \right\| = 1, \qquad \left\| A \begin{bmatrix} 0.785 \\ -0.620 \end{bmatrix} \right\| = \left\| \begin{bmatrix} 0.46 \\ 0.14 \\ -0.18 \end{bmatrix} \right\| = 0.514$$

for all $x \neq 0$, we have

$$0.514 \leq \frac{\|Ax\|}{\|x\|} \leq 9.53$$

Properties of matrix norm

- consistent with vector norm: matrix norm of $a \in \mathbf{R}^{n\times 1}$ is $\sqrt{\lambda_{\max}(a^Ta)} = \sqrt{a^Ta}$
- for any x, $\|Ax\| \leq \|A\|\|x\|$
- scaling: $\|aA\| = |a|\|A\|$
- triangle inequality: $\|A+B\| \leq \|A\| + \|B\|$
- definiteness: $\|A\| = 0 \quad \Leftrightarrow \quad A = 0$
- norm of product: $\|AB\| \leq \|A\|\|B\|$

Singular value decomposition

more complete picture of gain properties of A given by *singular value decomposition* (SVD) of A:

$$A = U\Sigma V^T$$

where

- $A \in \mathbf{R}^{m\times n}$, $\mathbf{Rank}(A) = r$
- $U \in \mathbf{R}^{m\times r}$, $U^TU = I$
- $V \in \mathbf{R}^{n\times r}$, $V^TV = I$
- $\Sigma = \mathbf{diag}(\sigma_1, \ldots, \sigma_r)$, where $\sigma_1 \geq \cdots \geq \sigma_r > 0$

with $U = [u_1 \cdots u_r]$, $V = [v_1 \cdots v_r]$,

$$A = U\Sigma V^T = \sum_{i=1}^{r} \sigma_i u_i v_i^T$$

- σ_i are the (nonzero) *singular values* of A
- v_i are the *right* or *input singular vectors* of A
- u_i are the *left* or *output singular vectors* of A

$$A^TA = (U\Sigma V^T)^T(U\Sigma V^T) = V\Sigma^2V^T$$

hence:

- v_i are eigenvectors of A^TA (corresponding to nonzero eigenvalues)
- $\sigma_i = \sqrt{\lambda_i(A^TA)}$ (and $\lambda_i(A^TA) = 0$ for $i > r$)
- $\|A\| = \sigma_1$

similarly,

$$AA^T = (U\Sigma V^T)(U\Sigma V^T)^T = U\Sigma^2U^T$$

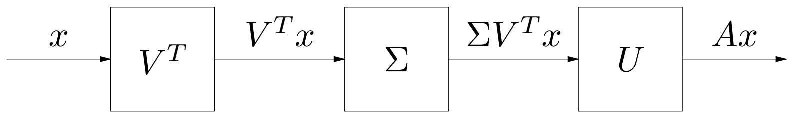
hence:

- u_i are eigenvectors of AA^T (corresponding to nonzero eigenvalues)
- $\sigma_i = \sqrt{\lambda_i(AA^T)}$ (and $\lambda_i(AA^T) = 0$ for $i > r$)

- $u_1, \ldots u_r$ are orthonormal basis for $\mathrm{range}(A)$
- $v_1, \ldots v_r$ are orthonormal basis for $\mathcal{N}(A)^\perp$

Interpretations

$$A = U\Sigma V^T = \sum_{i=1}^{r} \sigma_i u_i v_i^T$$



linear mapping $y = Ax$ can be decomposed as

- compute coefficients of x along input directions $v_1, \ldots, v_r$
- scale coefficients by σ_i
- reconstitute along output directions $u_1, \ldots, u_r$

difference with eigenvalue decomposition for symmetric A: input and output directions are *different*

- v_1 is most sensitive (highest gain) input direction
- u_1 is highest gain output direction
- $Av_1 = \sigma_1 u_1$

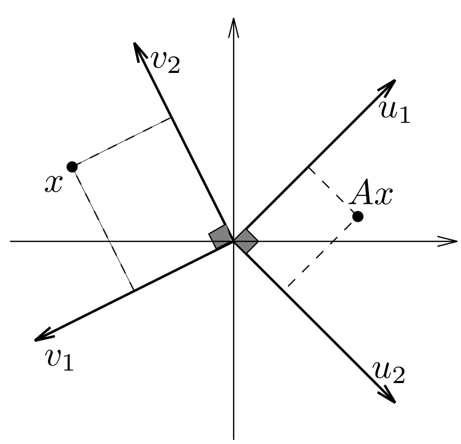
SVD gives clearer picture of gain as function of input/output directions

example: consider $A \in \mathbf{R}^{4\times 4}$ with $\Sigma = \mathbf{diag}(10,\ 7,\ 0.1,\ 0.05)$

- input components along directions v_1 and v_2 are amplified (by about 10) and come out mostly along plane spanned by u_1, u_2
- input components along directions v_3 and v_4 are attenuated (by about 10)
- $\|Ax\|/\|x\|$ can range between 10 and 0.05
- A is nonsingular
- for some applications you might say A is *effectively* rank 2

example: $A \in \mathbf{R}^{2\times 2}$, with $\sigma_1 = 1$, $\sigma_2 = 0.5$

- resolve x along $v_1,\ v_2$: $v_1^T x = 0.5$, $v_2^T x = 0.6$, *i.e.*, $x = 0.5v_1 + 0.6v_2$
- now form $Ax = (v_1^T x)\sigma_1 u_1 + (v_2^T x)\sigma_2 u_2 = (0.5)(1)u_1 + (0.6)(0.5)u_2$



Lecture 16
SVD Applications

- general pseudo-inverse
- full SVD
- image of unit ball under linear transformation
- SVD in estimation/inversion
- sensitivity of linear equations to data error
- low rank approximation via SVD

General pseudo-inverse

if $A \neq 0$ has SVD $A = U\Sigma V^T$,

$$A^\dagger = V\Sigma^{-1}U^T$$

is the *pseudo-inverse* or *Moore-Penrose inverse* of A

if A is skinny and full rank,

$$A^\dagger = (A^TA)^{-1}A^T$$

gives the least-squares approximate solution $x_{\rm ls} = A^\dagger y$

if A is fat and full rank,

$$A^\dagger = A^T(AA^T)^{-1}$$

gives the least-norm solution $x_{\rm ln} = A^\dagger y$

in general case:

$$X_{\rm ls} = \{ \ z \mid \|Az - y\| = \min_w \ \|Aw - y\| \ \}$$

is set of least-squares approximate solutions

$x_{\rm pinv} = A^\dagger y \in X_{\rm ls}$ has minimum norm on $X_{\rm ls}$, *i.e.*, $x_{\rm pinv}$ is the minimum-norm, least-squares approximate solution

Pseudo-inverse via regularization

for $\mu > 0$, let x_μ be (unique) minimizer of

$$\|Ax - y\|^2 + \mu \|x\|^2$$

i.e.,

$$x_\mu = \left(A^TA + \mu I\right)^{-1} A^T y$$

here, $A^TA + \mu I > 0$ and so is invertible

then we have $\lim_{\mu \to 0} x_\mu = A^\dagger y$

in fact, we have $\lim_{\mu \to 0} \left(A^TA + \mu I\right)^{-1} A^T = A^\dagger$

(check this!)

Full SVD

SVD of $A \in \mathbf{R}^{m \times n}$ with $\mathbf{Rank}(A) = r$:

$$A = U_1 \Sigma_1 V_1^T = \left[\begin{array}{ccc} u_1 & \cdots & u_r \end{array} \right] \left[\begin{array}{ccc} \sigma_1 & & \\ & \ddots & \\ & & \sigma_r \end{array} \right] \left[\begin{array}{c} v_1^T \\ \vdots \\ v_r^T \end{array} \right]$$

- find $U_2 \in \mathbf{R}^{m \times (m-r)}$, $V_2 \in \mathbf{R}^{n \times (n-r)}$ s.t. $U = [U_1 \ U_2] \in \mathbf{R}^{m \times m}$ and $V = [V_1 \ V_2] \in \mathbf{R}^{n \times n}$ are orthogonal
- add zero rows/cols to Σ_1 to form $\Sigma \in \mathbf{R}^{m \times n}$:

$$\Sigma = \left[\begin{array}{c|c} \Sigma_1 & 0_{r \times (n-r)} \\ \hline 0_{(m-r) \times r} & 0_{(m-r) \times (n-r)} \end{array} \right]$$

then we have

$$A = U_1 \Sigma_1 V_1^T = \left[\begin{array}{c|c} U_1 & U_2 \end{array} \right] \left[\begin{array}{c|c} \Sigma_1 & 0_{r \times (n-r)} \\ \hline 0_{(m-r) \times r} & 0_{(m-r) \times (n-r)} \end{array} \right] \left[\begin{array}{c} V_1^T \\ \hline V_2^T \end{array} \right]$$

i.e.:

$$A = U \Sigma V^T$$

called *full SVD* of A

(SVD with positive singular values only called *compact SVD*)

Image of unit ball under linear transformation

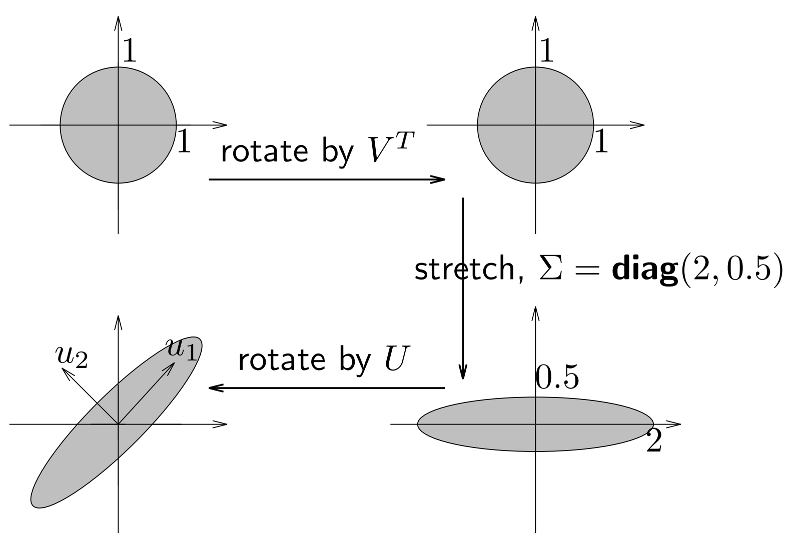
full SVD:

$$A = U\Sigma V^T$$

gives intepretation of $y = Ax$:

- rotate (by V^T)
- stretch along axes by σ_i ($\sigma_i = 0$ for $i > r$)
- zero-pad (if $m > n$) or truncate (if $m < n$) to get m-vector
- rotate (by U)

Image of unit ball under A



$\{Ax \mid \|x\| \le 1\}$ is *ellipsoid* with principal axes $\sigma_i u_i$.

SVD in estimation/inversion

suppose $y = Ax + v$, where

- $y \in \mathbf{R}^m$ is measurement
- $x \in \mathbf{R}^n$ is vector to be estimated
- v is a measurement noise or error

'norm-bound' model of noise: we assume $\|v\| \leq \alpha$ but otherwise know nothing about v (α gives max norm of noise)

- consider estimator $\hat{x} = By$, with $BA = I$ $(i.e.$, unbiased)
- estimation or inversion error is $\tilde{x} = \hat{x} - x = Bv$
- set of possible estimation errors is ellipsoid

$$\tilde{x} \in \mathcal{E}_{\rm unc} = \{\ Bv \mid \|v\| \leq \alpha\ \}$$

- $x = \hat{x} - \tilde{x} \in \hat{x} - \mathcal{E}_{\rm unc} = \hat{x} + \mathcal{E}_{\rm unc}$, $i.e.$:
  true x lies in *uncertainty ellipsoid* $\mathcal{E}_{\rm unc}$, centered at estimate $\hat{x}$
- 'good' estimator has 'small' $\mathcal{E}_{\rm unc}$ (with $BA = I$, of course)

semiaxes of $\mathcal{E}_{\rm unc}$ are $\alpha\sigma_i u_i$ (singular values & vectors of B)

e.g., maximum norm of error is $\alpha\|B\|$, *i.e.*, $\|\hat{x} - x\| \le \alpha\|B\|$

optimality of least-squares: suppose $BA = I$ is any estimator, and $B_{\rm ls} = A^\dagger$ is the least-squares estimator

then:

- $B_{\rm ls}B_{\rm ls}^T \le BB^T$
- $\mathcal{E}_{\rm ls} \subseteq \mathcal{E}$
- in particular $\|B_{\rm ls}\| \le \|B\|$

i.e., the least-squares estimator gives the *smallest* uncertainty ellipsoid

Example: navigation using range measurements (lect. 4)

we have

$$y = -\begin{bmatrix} k_1^T \\ k_2^T \\ k_3^T \\ k_4^T \end{bmatrix} x$$

where $k_i \in \mathbf{R}^2$

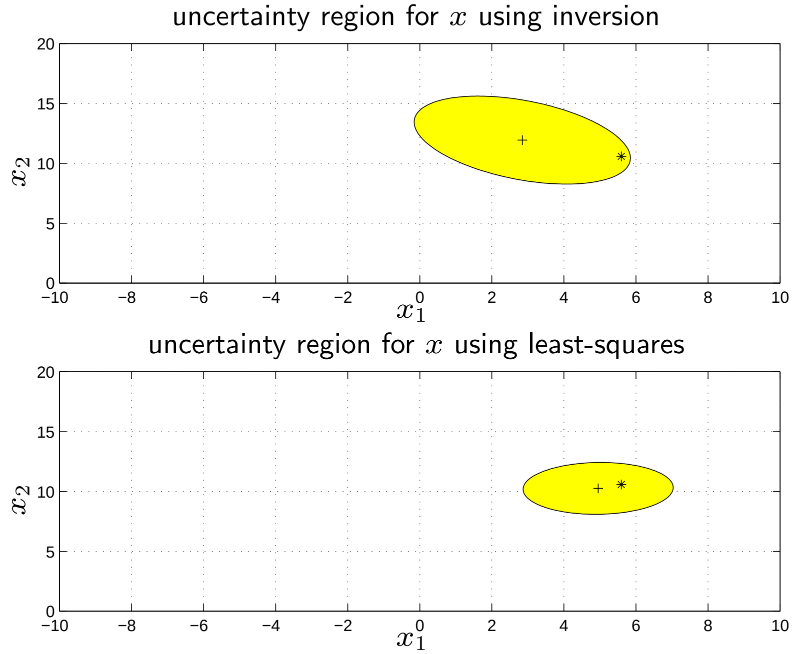
using first two measurements and inverting:

$$\hat{x} = -\left[\begin{bmatrix} k_1^T \\ k_2^T \end{bmatrix}^{-1} \quad 0_{2\times 2} \right] y$$

using all four measurements and least-squares:

$$\hat{x} = A^\dagger y$$

uncertainty regions (with $\alpha = 1$):



Proof of optimality property

suppose $A \in \mathbf{R}^{m \times n}$, $m > n$, is full rank

SVD: $A = U\Sigma V^T$, with V orthogonal

$B_{\rm ls} = A^\dagger = V\Sigma^{-1}U^T$, and B satisfies $BA = I$

define $Z = B - B_{\rm ls}$, so $B = B_{\rm ls} + Z$

then $ZA = ZU\Sigma V^T = 0$, so $ZU = 0$ (multiply by $V\Sigma^{-1}$ on right)

therefore

$$\begin{aligned} BB^T &= (B_{\rm ls} + Z)(B_{\rm ls} + Z)^T \\ &= B_{\rm ls}B_{\rm ls}^T + B_{\rm ls}Z^T + ZB_{\rm ls}^T + ZZ^T \\ &= B_{\rm ls}B_{\rm ls}^T + ZZ^T \\ &\geq B_{\rm ls}B_{\rm ls}^T \end{aligned}$$

using $ZB_{\rm ls}^T = (ZU)\Sigma^{-1}V^T = 0$

Sensitivity of linear equations to data error

consider $y = Ax$, $A \in \mathbf{R}^{n \times n}$ invertible; of course $x = A^{-1}y$

suppose we have an error or noise in y, $i.e.$, y becomes $y + \delta y$

then x becomes $x + \delta x$ with $\delta x = A^{-1}\delta y$

hence we have $\|\delta x\| = \|A^{-1}\delta y\| \leq \|A^{-1}\|\|\delta y\|$

if $\|A^{-1}\|$ is large,

- small errors in y can lead to large errors in x
- can't solve for x given y (with small errors)
- hence, A can be considered singular in practice

a more refined analysis uses *relative* instead of *absolute* errors in x and y

since $y = Ax$, we also have $\|y\| \leq \|A\|\|x\|$, hence

$$\frac{\|\delta x\|}{\|x\|} \leq \|A\|\|A^{-1}\|\frac{\|\delta y\|}{\|y\|}$$

$$\kappa(A) = \|A\|\|A^{-1}\| = \sigma_{\max}(A)/\sigma_{\min}(A)$$

is called the *condition number* of A

we have:

relative error in solution $x \leq$ condition number $\cdot$ relative error in data y

or, in terms of # bits of guaranteed accuracy:

bits accuracy in solution $\approx$ # bits accuracy in data $- \log_2 \kappa$

we say

- A is well conditioned if κ is small
- A is poorly conditioned if κ is large

(definition of 'small' and 'large' depend on application)

same analysis holds for least-squares approximate solutions with A nonsquare, $\kappa = \sigma_{\max}(A)/\sigma_{\min}(A)$

Low rank approximations

suppose $A \in \mathbf{R}^{m\times n}$, $\mathbf{Rank}(A) = r$, with SVD $A = U\Sigma V^T = \sum_{i=1}^{r} \sigma_i u_i v_i^T$

we seek matrix $\hat{A}$, $\mathbf{Rank}(\hat{A}) \leq p < r$, s.t. $\hat{A} \approx A$ in the sense that $\|A - \hat{A}\|$ is minimized

solution: optimal rank p approximator is

$$\hat{A} = \sum_{i=1}^{p} \sigma_i u_i v_i^T$$

- hence $\|A - \hat{A}\| = \left\| \sum_{i=p+1}^{r} \sigma_i u_i v_i^T \right\| = \sigma_{p+1}$
- interpretation: SVD dyads $u_i v_i^T$ are ranked in order of 'importance'; take p to get rank p approximant

proof: suppose $\mathbf{Rank}(B) \leq p$

then $\dim \mathcal{N}(B) \geq n - p$

also, $\dim \operatorname{span}\{v_1, \ldots, v_{p+1}\} = p + 1$

hence, the two subspaces intersect, *i.e.*, there is a unit vector $z \in \mathbf{R}^n$ s.t.

$$Bz = 0, \qquad z \in \operatorname{span}\{v_1, \ldots, v_{p+1}\}$$

$$(A - B)z = Az = \sum_{i=1}^{p+1} \sigma_i u_i v_i^T z$$

$$\|(A - B)z\|^2 = \sum_{i=1}^{p+1} \sigma_i^2 (v_i^T z)^2 \geq \sigma_{p+1}^2 \|z\|^2$$

hence $\|A - B\| \geq \sigma_{p+1} = \|A - \hat{A}\|$

Distance to singularity

another interpretation of σ_i:

$$\sigma_i = \min\{ \ \|A - B\| \ | \ \mathbf{Rank}(B) \leq i - 1 \ \}$$

i.e., the distance (measured by matrix norm) to the nearest rank $i - 1$ matrix

for example, if $A \in \mathbf{R}^{n \times n}$, $\sigma_n = \sigma_{\min}$ is distance to nearest singular matrix

hence, small $\sigma_{\min}$ means A is near to a singular matrix

application: model simplification

suppose $y = Ax + v$, where

- $A \in \mathbf{R}^{100\times 30}$ has SVs

$$10,\ 7,\ 2,\ 0.5,\ 0.01,\ \ldots, 0.0001$$

- $\|x\|$ is on the order of 1

- unknown error or noise v has norm on the order of 0.1

then the terms $\sigma_i u_i v_i^T x$, for $i = 5, \ldots, 30$, are substantially smaller than the noise term v

simplified model:

$$y = \sum_{i=1}^{4} \sigma_i u_i v_i^T x + v$$

Lecture 17
Example: Quantum mechanics

- wave function and Schrodinger equation
- discretization
- preservation of probability
- eigenvalues & eigenstates
- example

Quantum mechanics

- single particle in interval $[0, 1]$, mass m
- potential $V : [0, 1] \to \mathbf{R}$

$\Psi : [0, 1] \times \mathbf{R}_+ \to \mathbf{C}$ is (complex-valued) *wave function*

interpretation: $|\Psi(x, t)|^2$ is probability density of particle at position x, time t

(so $\int_0^1 |\Psi(x, t)|^2 \, dx = 1$ for all t)

evolution of Ψ governed by *Schrodinger* equation:

$$i\hbar\dot{\Psi} = \left(V - \frac{\hbar^2}{2m}\nabla_x^2 \right) \Psi = H\Psi$$

where H is *Hamiltonian* operator, $i = \sqrt{-1}$

Discretization

let's discretize position x into N discrete points, k/N, $k = 1, \ldots, N$

wave function is approximated as *vector* $\Psi(t) \in \mathbf{C}^N$

∇_x^2 operator is approximated as *matrix*

$$\nabla^2 = N^2 \begin{bmatrix} -2 & 1 & & & & \\ 1 & -2 & 1 & & & \\ & 1 & -2 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & & 1 & -2 \end{bmatrix}$$

so $w = \nabla^2 v$ means

$$w_k = \frac{(v_{k+1} - v_k)/(1/N) - (v_k - v_{k-1})(1/N)}{1/N}$$

(which approximates $w = \partial^2 v / \partial x^2$)

discretized Schrodinger equation is (complex) linear dynamical system

$$\dot{\Psi} = (-i/\hbar)(V - (\hbar/2m)\nabla^2)\Psi = (-i/\hbar)H\Psi$$

where V is a diagonal matrix with $V_{kk} = V(k/N)$

hence we analyze using linear dynamical system theory (with complex vectors & matrices):

$$\dot{\Psi} = (-i/\hbar)H\Psi$$

solution of Shrodinger equation: $\Psi(t) = e^{(-i/\hbar)tH}\Psi(0)$

matrix $e^{(-i/\hbar)tH}$ propogates wave function forward in time t seconds (backward if $t < 0$)

Preservation of probability

$$\begin{aligned}
\frac{d}{dt}\|\Psi\|^2 &= \frac{d}{dt}\Psi^*\Psi \\
&= \dot{\Psi}^*\Psi + \Psi^*\dot{\Psi} \\
&= ((-i/\hbar)H\Psi)^*\Psi + \Psi^*((-i/\hbar)H\Psi) \\
&= (i/\hbar)\Psi^*H\Psi + (-i/\hbar)\Psi^*H\Psi \\
&= 0
\end{aligned}$$

(using $H = H^T \in \mathbf{R}^{N\times N}$)

hence, $\|\Psi(t)\|^2$ is constant; our discretization preserves probability *exactly*

$U = e^{-(i/\hbar)tH}$ is *unitary*, meaning $U^*U = I$

unitary is extension of *orthogonal* for complex matrix: if $U \in \mathbf{C}^{N\times N}$ is unitary and $z \in \mathbf{C}^N$, then

$$\|Uz\|^2 = (Uz)^*(Uz) = z^*U^*Uz = z^*z = \|z\|^2$$

Eigenvalues & eigenstates

H is symmetric, so

- its eigenvalues $\lambda_1, \ldots, \lambda_N$ are real ($\lambda_1 \leq \cdots \leq \lambda_N$)
- its eigenvectors $v_1, \ldots, v_N$ can be chosen to be orthogonal (and real)

from $Hv = \lambda v \;\Leftrightarrow\; (-i/\hbar)Hv = (-i/\hbar)\lambda v$ we see:

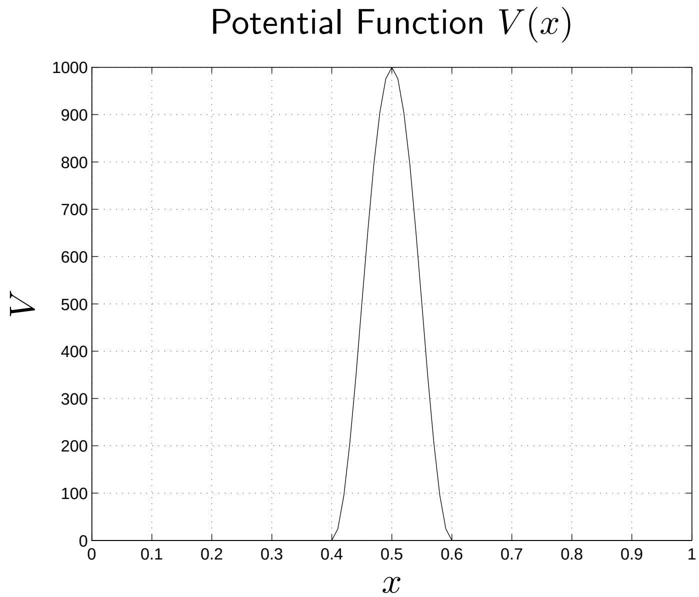
- eigenvectors of $(-i/\hbar)H$ are same as eigenvectors of H, $i.e.$, $v_1, \ldots, v_N$
- eigenvalues of $(-i/\hbar)H$ are $(-i/\hbar)\lambda_1, \ldots, (-i/\hbar)\lambda_N$ (which are pure imaginary)

- eigenvectors v_k are called *eigenstates* of system
- eigenvalue λ_k is *energy* of eigenstate v_k
- for mode $\Psi(t) = e^{(-i/\hbar)\lambda_k t} v_k$, probability density

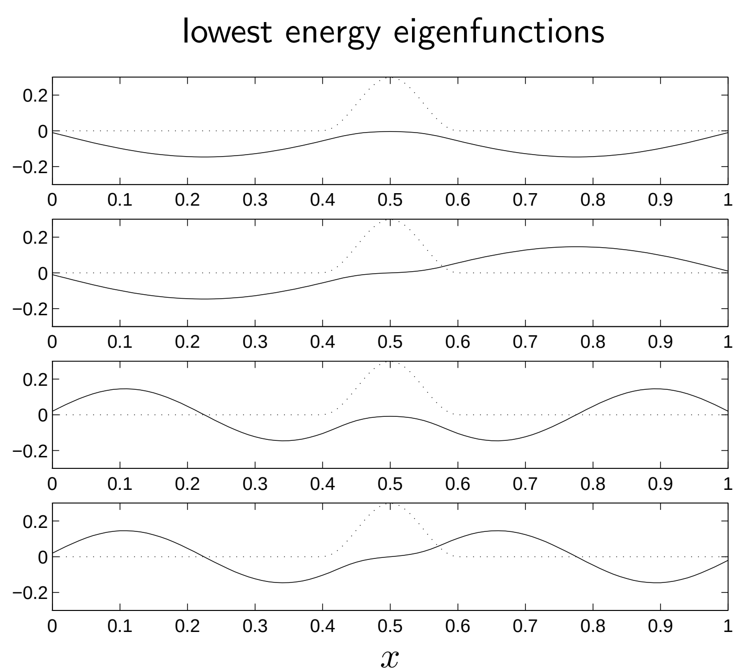
$$|\Psi_m(t)|^2 = \left| e^{(-i/\hbar)\lambda_k t} v_k \right|^2 = |v_{mk}|^2$$

doesn't change with time (v_{mk} is mth entry of v_k)

Example



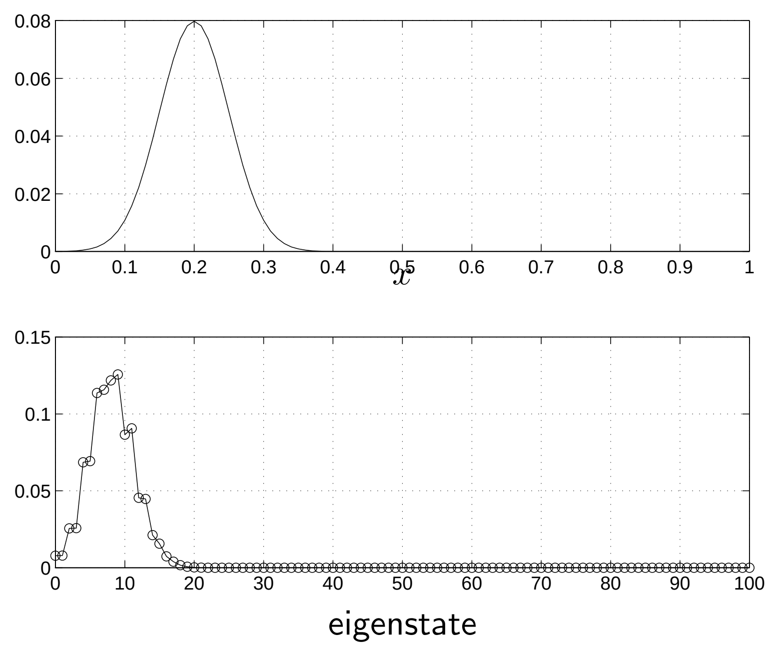
- potential bump in middle of infinite potential well
- (for this example, we set $\hbar = 1$, $m = 1$ . . .)



- potential V shown as dotted line (scaled to fit plot)
- four eigenstates with lowest energy shown ($i.e.$, v_1, v_2, v_3, v_4)

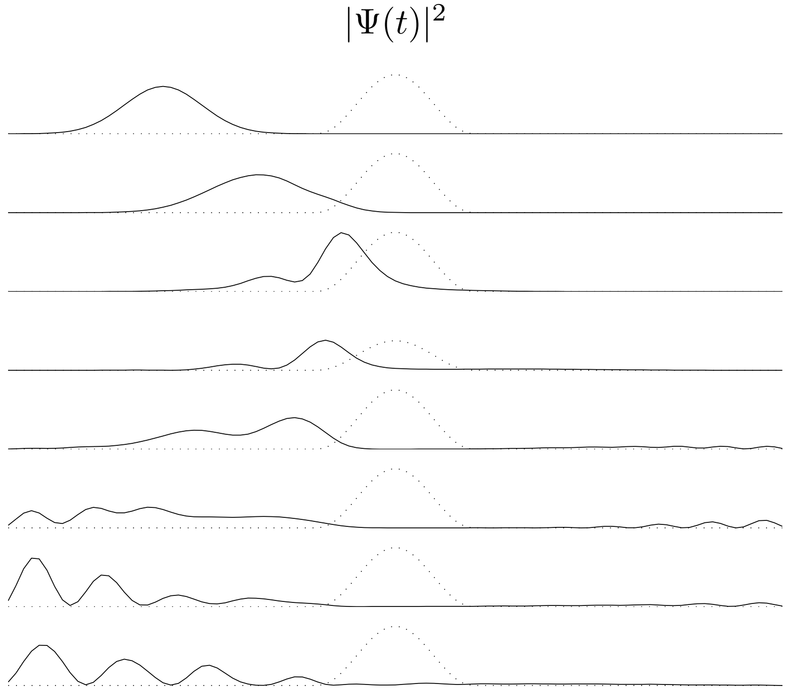
now let's look at a trajectory of Ψ, with initial wave function $\Psi(0)$

- particle near $x = 0.2$
- with momentum to right (can't see in plot of $|\Psi|^2$)
- (expected) kinetic energy half potential bump height



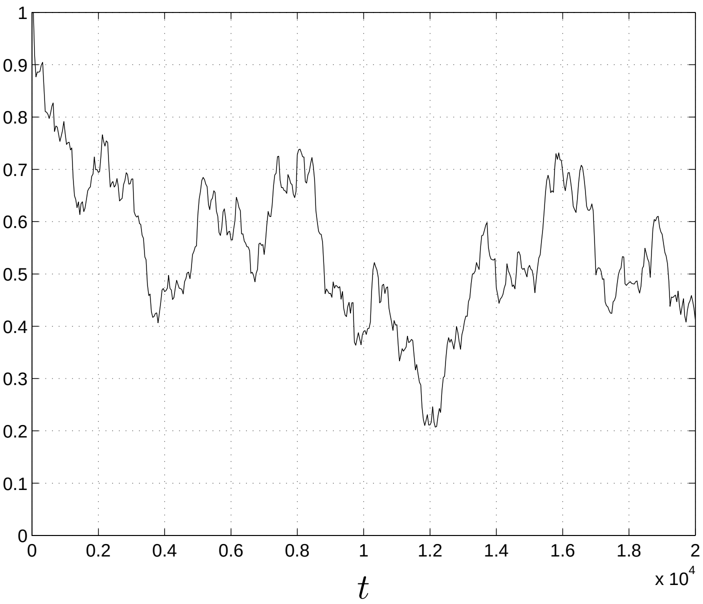
- top plot shows initial probability density $|\Psi(0)|^2$
- bottom plot shows $|v_k^* \Psi(0)|^2$, $i.e.$, resolution of $\Psi(0)$ into eigenstates

time evolution, for $t = 0, 40, 80, \ldots, 320$:



cf. classical solution:

- particle rolls half way up potential bump, stops, then rolls back down
- reverses velocity when it hits the wall at left
  (perfectly elastic collision)
- then repeats



plot shows probability that particle is in left half of well, *i.e.*, $\sum_{k=1}^{N/2} |\Psi_k(t)|^2$, versus time t

Lecture 18
Controllability and state transfer

- state transfer
- reachable set, controllability matrix
- minimum norm inputs
- infinite-horizon minimum norm transfer

State transfer

consider $\dot{x} = Ax + Bu$ (or $x(t+1) = Ax(t) + Bu(t)$) over time interval $[t_i, t_f]$

we say input $u : [t_i, t_f] \to \mathbf{R}^m$ *steers* or *transfers* state from $x(t_i)$ to $x(t_f)$ (over time interval $[t_i, t_f]$)

(subscripts stand for *initial* and *final*)

questions:

- where can $x(t_i)$ be transfered to at $t = t_f$?
- how quickly can $x(t_i)$ be transfered to some $x_{\rm target}$?
- how do we find a u that transfers $x(t_i)$ to $x(t_f)$?
- how do we find a 'small' or 'efficient' u that transfers $x(t_i)$ to $x(t_f)$?

Reachability

consider state transfer from $x(0) = 0$ to $x(t)$

we say $x(t)$ is *reachable* (in t seconds or epochs)

we define $\mathcal{R}_t \subseteq \mathbf{R}^n$ as the set of points reachable in t seconds or epochs

for CT system $\dot{x} = Ax + Bu$,

$$\mathcal{R}_t = \left\{ \int_0^t e^{(t-\tau)A} Bu(\tau)\, d\tau \;\middle|\; u : [0, t] \to \mathbf{R}^m \right\}$$

and for DT system $x(t+1) = Ax(t) + Bu(t)$,

$$\mathcal{R}_t = \left\{ \sum_{\tau=0}^{t-1} A^{t-1-\tau} Bu(\tau) \;\middle|\; u(t) \in \mathbf{R}^m \right\}$$

- $\mathcal{R}_t$ is a subspace of $\mathbf{R}^n$
- $\mathcal{R}_t \subseteq \mathcal{R}_s$ if $t \leq s$

  ($i.e.$, can reach more points given more time)

we define the *reachable set* $\mathcal{R}$ as the set of points reachable for some t:

$$\mathcal{R} = \bigcup_{t \geq 0} \mathcal{R}_t$$

Reachability for discrete-time LDS

DT system $x(t+1) = Ax(t) + Bu(t)$, $x(t) \in \mathbf{R}^n$

$$x(t) = \mathcal{C}_t \begin{bmatrix} u(t-1) \\ \vdots \\ u(0) \end{bmatrix}$$

where $\mathcal{C}_t = \begin{bmatrix} B & AB & \cdots & A^{t-1}B \end{bmatrix}$

so reachable set at t is $\mathcal{R}_t = \mathrm{range}(\mathcal{C}_t)$

by C-H theorem, we can express each A^k for $k \geq n$ as linear combination of $A^0, \ldots, A^{n-1}$

hence for $t \geq n$, $\mathrm{range}(\mathcal{C}_t) = \mathrm{range}(\mathcal{C}_n)$

thus we have

$$\mathcal{R}_t = \begin{cases} \mathrm{range}(\mathcal{C}_t) & t < n \\ \mathrm{range}(\mathcal{C}) & t \geq n \end{cases}$$

where $\mathcal{C} = \mathcal{C}_n$ is called the *controllability matrix*

- any state that can be reached can be reached by $t = n$
- the reachable set is $\mathcal{R} = \mathrm{range}(\mathcal{C})$

Controllable system

system is called *reachable* or *controllable* if all states are reachable ($i.e.$, $\mathcal{R} = \mathbf{R}^n$)

system is reachable if and only if $\mathbf{Rank}(\mathcal{C}) = n$

example: $x(t+1) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u(t)$

controllability matrix is $\mathcal{C} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$

hence system is not controllable; reachable set is

$$\mathcal{R} = \mathrm{range}(\mathcal{C}) = \{\ x \mid x_1 = x_2\ \}$$

General state transfer

with $t_f > t_i$,

$$x(t_f) = A^{t_f - t_i} x(t_i) + \mathcal{C}_{t_f - t_i} \begin{bmatrix} u(t_f - 1) \\ \vdots \\ u(t_i) \end{bmatrix}$$

hence can transfer $x(t_i)$ to $x(t_f) = x_{\mathrm{des}}$

$$\Leftrightarrow \qquad x_{\mathrm{des}} - A^{t_f - t_i} x(t_i) \in \mathcal{R}_{t_f - t_i}$$

- general state transfer reduces to reachability problem
- if system is controllable any state transfer can be achieved in $\leq n$ steps
- important special case: driving state to zero (sometimes called regulating or controlling state)

Least-norm input for reachability

assume system is reachable, $\mathbf{Rank}(\mathcal{C}_t) = n$

to steer $x(0) = 0$ to $x(t) = x_{\mathrm{des}}$, inputs $u(0), \ldots, u(t-1)$ must satisfy

$$x_{\mathrm{des}} = \mathcal{C}_t \begin{bmatrix} u(t-1) \\ \vdots \\ u(0) \end{bmatrix}$$

among all u that steer $x(0) = 0$ to $x(t) = x_{\mathrm{des}}$, the one that minimizes

$$\sum_{\tau=0}^{t-1} \|u(\tau)\|^2$$

is given by

$$\begin{bmatrix} u_{\mathrm{ln}}(t-1) \\ \vdots \\ u_{\mathrm{ln}}(0) \end{bmatrix} = \mathcal{C}_t^T (\mathcal{C}_t \mathcal{C}_t^T)^{-1} x_{\mathrm{des}}$$

u_{ln} is called *least-norm* or *minimum energy* input that effects state transfer

can express as

$$u_{\mathrm{ln}}(\tau) = B^T (A^T)^{(t-1-\tau)} \left(\sum_{s=0}^{t-1} A^s B B^T (A^T)^s \right)^{-1} x_{\mathrm{des}},$$

for $\tau = 0, \ldots, t-1$

$\mathcal{E}_{\min}$, the minimum value of $\sum_{\tau=0}^{t-1} \|u(\tau)\|^2$ required to reach $x(t) = x_{\rm des}$, is sometimes called *minimum energy* required to reach $x(t) = x_{\rm des}$

$$
\begin{aligned}
\mathcal{E}_{\min} &= \sum_{\tau=0}^{t-1} \|u_{\rm ln}(\tau)\|^2 \\
&= \left(\mathcal{C}_t^T(\mathcal{C}_t\mathcal{C}_t^T)^{-1}x_{\rm des}\right)^T \mathcal{C}_t^T(\mathcal{C}_t\mathcal{C}_t^T)^{-1}x_{\rm des} \\
&= x_{\rm des}^T(\mathcal{C}_t\mathcal{C}_t^T)^{-1}x_{\rm des} \\
&= x_{\rm des}^T \left(\sum_{\tau=0}^{t-1} A^\tau BB^T(A^T)^\tau\right)^{-1} x_{\rm des}
\end{aligned}
$$

- $\mathcal{E}_{\min}(x_{\rm des}, t)$ gives measure of how hard it is to reach $x(t) = x_{\rm des}$ from $x(0) = 0$ ($i.e.$, how large a u is required)
- $\mathcal{E}_{\min}(x_{\rm des}, t)$ gives practical measure of controllability/reachability (as function of $x_{\rm des}$, t)
- ellipsoid $\{ z \mid \mathcal{E}_{\min}(z, t) \leq 1 \}$ shows points in state space reachable at t with one unit of energy

  (shows directions that can be reached with small inputs, and directions that can be reached only with large inputs)

$\mathcal{E}_{\min}$ as function of t:

if $t \geq s$ then
$$\sum_{\tau=0}^{t-1} A^\tau BB^T(A^T)^\tau \geq \sum_{\tau=0}^{s-1} A^\tau BB^T(A^T)^\tau$$

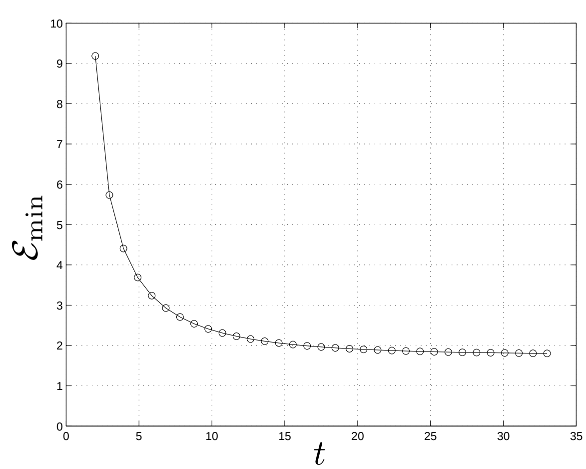
hence
$$\left(\sum_{\tau=0}^{t-1} A^\tau BB^T(A^T)^\tau\right)^{-1} \leq \left(\sum_{\tau=0}^{s-1} A^\tau BB^T(A^T)^\tau\right)^{-1}$$

so $\mathcal{E}_{\min}(x_{\text{des}}, t) \leq \mathcal{E}_{\min}(x_{\text{des}}, s)$

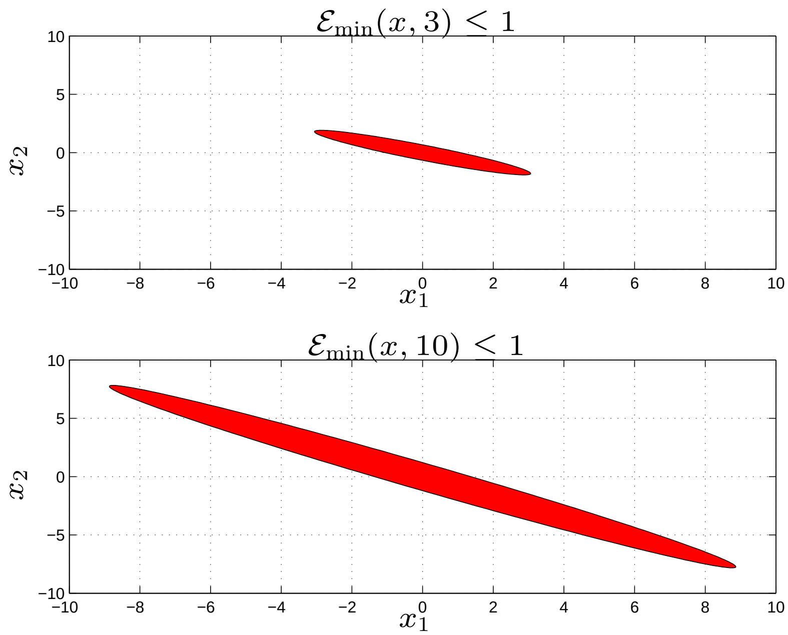
$i.e.$: takes less energy to get somewhere more leisurely

example: $x(t+1) = \begin{bmatrix} 1.75 & 0.8 \\ -0.95 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u(t)$

$\mathcal{E}_{\min}(z, t)$ for $z = [1\ 1]^T$:



ellipsoids $\mathcal{E}_{\min} \leq 1$ for $t = 3$ and $t = 10$:



Minimum energy over infinite horizon

the matrix

$$P = \lim_{t \to \infty} \left(\sum_{\tau=0}^{t-1} A^\tau B B^T (A^T)^\tau \right)^{-1}$$

always exists, and gives the minimum energy required to reach a point x_{des} (with no limit on t):

$$\min \left\{ \sum_{\tau=0}^{t-1} \|u(\tau)\|^2 \;\middle|\; x(0) = 0,\; x(t) = x_{\mathrm{des}} \right\} = x_{\mathrm{des}}^T P x_{\mathrm{des}}$$

if A is stable, $P > 0$ ($i.e.$, can't get anywhere for free)

if A is not stable, then P can have nonzero nullspace

- $Pz = 0$, $z \neq 0$ means can get to z using u's with energy as small as you like

  (u just gives a little kick to the state; the instability carries it out to z efficiently)

- basis of highly maneuverable, unstable aircraft

Continuous-time reachability

consider now $\dot{x} = Ax + Bu$ with $x(t) \in \mathbf{R}^n$

reachable set at time t is

$$\mathcal{R}_t = \left\{ \int_0^t e^{(t-\tau)A} Bu(\tau)\, d\tau \;\middle|\; u : [0,t] \to \mathbf{R}^m \right\}$$

fact: for $t > 0$, $\mathcal{R}_t = \mathcal{R} = \mathrm{range}(\mathcal{C})$, where

$$\mathcal{C} = \left[\begin{array}{cccc} B & AB & \cdots & A^{n-1}B \end{array} \right]$$

is the controllability matrix of (A, B)

- same $\mathcal{R}$ as discrete-time system

- for continuous-time system, any reachable point can be reached as fast as you like (with large enough u)

first let's show for *any* u (and $x(0) = 0$) we have $x(t) \in \mathrm{range}(\mathcal{C})$

write e^{tA} as power series:

$$e^{tA} = I + \frac{t}{1!}A + \frac{t^2}{2!}A^2 + \cdots$$

by C-H, express $A^n, A^{n+1}, \ldots$ in terms of $A^0, \ldots, A^{n-1}$ and collect powers of A:

$$e^{tA} = \alpha_0(t)I + \alpha_1(t)A + \cdots + \alpha_{n-1}(t)A^{n-1}$$

therefore

$$\begin{aligned} x(t) &= \int_0^t e^{\tau A} B u(t-\tau)\, d\tau \\ &= \int_0^t \left(\sum_{i=0}^{n-1} \alpha_i(\tau) A^i \right) B u(t-\tau)\, d\tau \end{aligned}$$

$$\begin{aligned} &= \sum_{i=0}^{n-1} A^i B \int_0^t \alpha_i(\tau) u(t-\tau)\, d\tau \\ &= \mathcal{C} z \end{aligned}$$

where $z_i = \displaystyle\int_0^t \alpha_i(\tau) u(t-\tau)\, d\tau$

hence, $x(t)$ is always in $\mathrm{range}(\mathcal{C})$

need to show converse: every point in $\mathrm{range}(\mathcal{C})$ can be reached

Impulsive inputs

suppose $x(0_-) = 0$ and we apply input $u(t) = \delta^{(k)}(t)f$, where $\delta^{(k)}$ denotes kth derivative of δ and $f \in \mathbf{R}^m$

then $U(s) = s^k f$, so

$$\begin{aligned} X(s) &= (sI - A)^{-1}Bs^k f \\ &= \left(s^{-1}I + s^{-2}A + \cdots\right) Bs^k f \\ &= (\underbrace{s^{k-1} + \cdots + sA^{k-2} + A^{k-1}}_{\text{impulsive terms}} + s^{-1}A^k + \cdots)Bf \end{aligned}$$

hence

$$x(t) = \text{ impulsive terms } + A^k Bf + A^{k+1}Bf\frac{t}{1!} + A^{k+2}Bf\frac{t^2}{2!} + \cdots$$

in particular, $x(0_+) = A^k Bf$

thus, input $u = \delta^{(k)}f$ transfers state from $x(0_-) = 0$ to $x(0_+) = A^k Bf$

now consider input of form

$$u(t) = \delta(t)f_0 + \cdots + \delta^{(n-1)}(t)f_{n-1}$$

where $f_i \in \mathbf{R}^m$

by linearity we have

$$x(0_+) = Bf_0 + \cdots + A^{n-1}Bf_{n-1} = \mathcal{C}\begin{bmatrix} f_0 \\ \vdots \\ f_{n-1} \end{bmatrix}$$

hence we can reach any point in $\mathrm{range}(\mathcal{C})$

(at least, using impulse inputs)

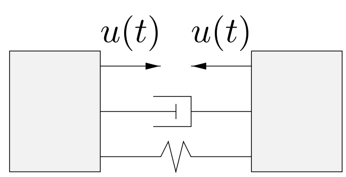
can also be shown that any point in $\mathrm{range}(\mathcal{C})$ can be reached for any $t > 0$ using *nonimpulsive* inputs

fact: if $x(0) \in \mathcal{R}$, then $x(t) \in \mathcal{R}$ for all t (no matter what u is)

to show this, need to show $e^{tA}x(0) \in \mathcal{R}$ if $x(0) \in \mathcal{R}$ . . .

Example

- unit masses at y_1, y_2, connected by unit springs, dampers
- input is tension between masses
- state is $x = [y^T \ \dot{y}^T]^T$



system is

$$\dot{x} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 1 \\ -1 \end{bmatrix} u$$

- can we maneuver state anywhere, starting from $x(0) = 0$?
- if not, where can we maneuver state?

controllability matrix is

$$\mathcal{C} = \begin{bmatrix} B & AB & A^2B & A^3B \end{bmatrix} = \begin{bmatrix} 0 & 1 & -2 & 2 \\ 0 & -1 & 2 & -2 \\ 1 & -2 & 2 & 0 \\ -1 & 2 & -2 & 0 \end{bmatrix}$$

hence reachable set is

$$\mathcal{R} = \operatorname{span}\left\{ \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ -1 \end{bmatrix} \right\}$$

we can reach states with $y_1 = -y_2$, $\dot{y}_1 = -\dot{y}_2$, *i.e.*, precisely the differential motions

it's obvious — internal force does not affect center of mass position or total momentum!

Least-norm input for reachability

(also called *minimum energy input*)

assume that $\dot{x} = Ax + Bu$ is reachable

we seek u that steers $x(0) = 0$ to $x(t) = x_{\text{des}}$ and minimizes

$$\int_0^t \|u(\tau)\|^2 \, d\tau$$

let's discretize system with interval $h = t/N$

(we'll let $N \to \infty$ later)

thus u is piecewise constant:

$$u(\tau) = u_d(k) \quad \text{for} \quad kh \le \tau < (k+1)h, \qquad k = 0, \ldots, N-1$$

so

$$x(t) = \left[\begin{array}{cccc} B_d & A_dB_d & \cdots & A_d^{N-1}B_d \end{array} \right] \left[\begin{array}{c} u_d(N-1) \\ \vdots \\ u_d(0) \end{array} \right]$$

where

$$A_d = e^{hA}, \quad B_d = \int_0^h e^{\tau A}\, d\tau B$$

least-norm u_d that yields $x(t) = x_{\rm des}$ is

$$u_{\rm dln}(k) = B_d^T (A_d^T)^{(N-1-k)} \left(\sum_{i=0}^{N-1} A_d^i B_d B_d^T (A_d^T)^i \right)^{-1} x_{\rm des}$$

let's express in terms of A:

$$B_d^T (A_d^T)^{(N-1-k)} = B_d^T e^{(t-\tau)A^T}$$

where $\tau = t(k+1)/N$

for N large, $B_d \approx (t/N)B$, so this is approximately

$$(t/N)B^T e^{(t-\tau)A^T}$$

similarly

$$\begin{aligned} \sum_{i=0}^{N-1} A_d^i B_d B_d^T (A_d^T)^i &= \sum_{i=0}^{N-1} e^{(ti/N)A} B_d B_d^T e^{(ti/N)A^T} \\ &\approx (t/N) \int_0^t e^{\bar{t}A} B B^T e^{\bar{t}A^T}\, d\bar{t} \end{aligned}$$

for large N

hence least-norm discretized input is approximately

$$u_{\text{ln}}(\tau) = B^T e^{(t-\tau)A^T} \left(\int_0^t e^{\bar{t}A} B B^T e^{\bar{t}A^T} \, d\bar{t} \right)^{-1} x_{\text{des}}, \quad 0 \le \tau \le t$$

for large N

hence, this is the least-norm continuous input

- can make t small, but get larger u
- cf. DT solution: sum becomes integral

min energy is

$$\int_0^t \|u_{\text{ln}}(\tau)\|^2 \, d\tau = x_{\text{des}}^T Q(t)^{-1} x_{\text{des}}$$

where

$$Q(t) = \int_0^t e^{\tau A} B B^T e^{\tau A^T} \, d\tau$$

can show

$$\begin{aligned} (A,B) \text{ controllable} \quad &\Leftrightarrow \quad Q(t) > 0 \text{ for all } t > 0 \\ &\Leftrightarrow \quad Q(s) > 0 \text{ for some } s > 0 \end{aligned}$$

in fact, $\text{range}(Q(t)) = \mathcal{R}$ for any $t > 0$

Minimum energy over infinite horizon

the matrix

$$P = \lim_{t\to\infty}\left(\int_0^t e^{\tau A}BB^Te^{\tau A^T}\,d\tau\right)^{-1}$$

always exists, and gives minimum energy required to reach a point $x_{\rm des}$ (with no limit on t):

$$\min\left\{\ \int_0^t \|u(\tau)\|^2\,d\tau\ \middle|\ x(0)=0,\ x(t)=x_{\rm des}\ \right\} = x_{\rm des}^T P x_{\rm des}$$

- if A is stable, $P>0$ ($i.e.$, can't get anywhere for free)
- if A is not stable, then P can have nonzero nullspace
- $Pz=0$, $z\neq 0$ means can get to z using u's with energy as small as you like (u just gives a little kick to the state; the instability carries it out to z efficiently)

General state transfer

consider state transfer from $x(t_i)$ to $x(t_f)=x_{\rm des}$, $t_f>t_i$

since

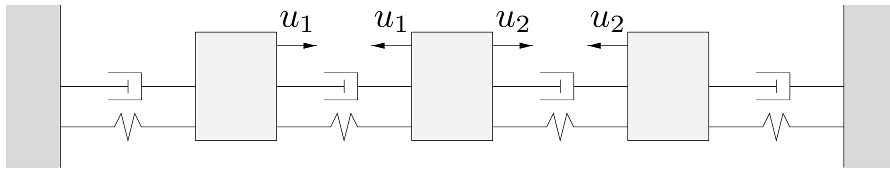
$$x(t_f) = e^{(t_f-t_i)A}x(t_i) + \int_{t_i}^{t_f} e^{(t_f-\tau)A}Bu(\tau)\,d\tau$$

u steers $x(t_i)$ to $x(t_f)=x_{\rm des}$ $\Leftrightarrow$

u (shifted by t_i) steers $x(0)=0$ to $x(t_f-t_i)=x_{\rm des}-e^{(t_f-t_i)A}x(t_i)$

- general state transfer reduces to reachability problem
- if system is controllable, any state transfer can be effected
  - in 'zero' time with impulsive inputs
  - in any positive time with non-impulsive inputs

Example



- unit masses, springs, dampers
- u_1 is force between 1st & 2nd masses
- u_2 is force between 2nd & 3rd masses
- $y \in \mathbf{R}^3$ is displacement of masses 1,2,3
- $x = \begin{bmatrix} y \\ \dot{y} \end{bmatrix}$

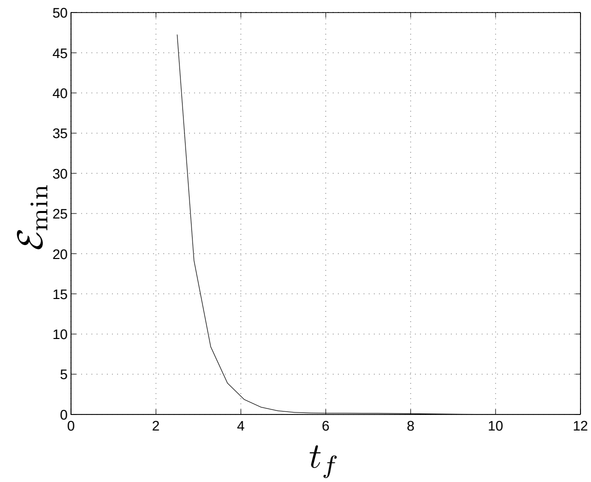
system is:

$$\dot{x} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -2 & 1 & 0 & -2 & 1 & 0 \\ 1 & -2 & 1 & 1 & -2 & 1 \\ 0 & 1 & -2 & 0 & 1 & -2 \end{bmatrix} x + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ -1 & 1 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

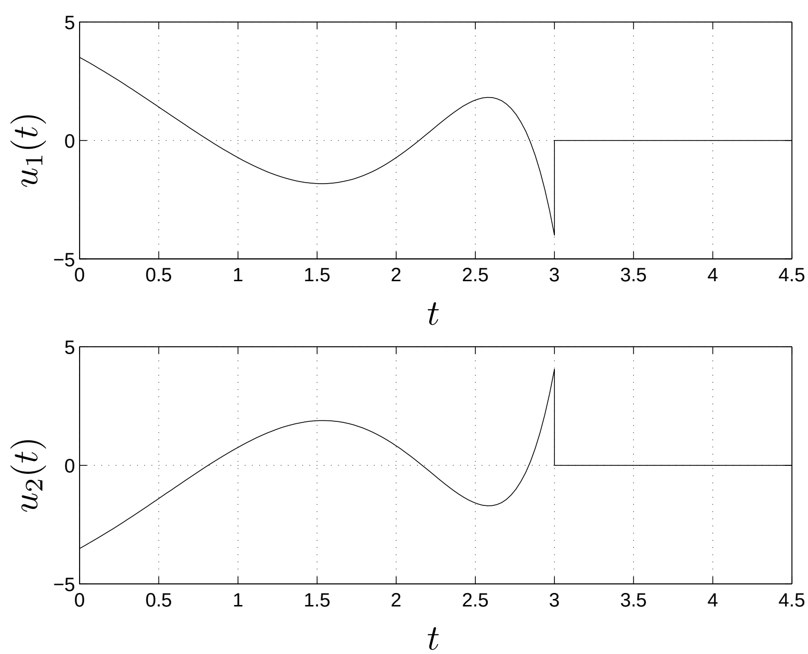
steer state from $x(0) = e_1$ to $x(t_f) = 0$

i.e., control initial state e_1 to zero at $t = t_f$

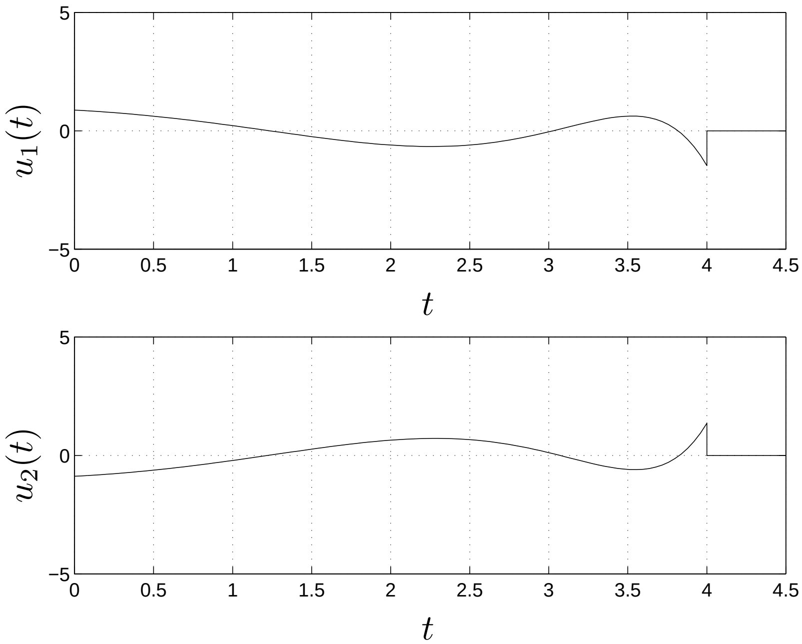
$\mathcal{E}_{\min} = \int_0^{t_f} \|u_{\rm ln}(\tau)\|^2 \, d\tau$ vs. t_f:



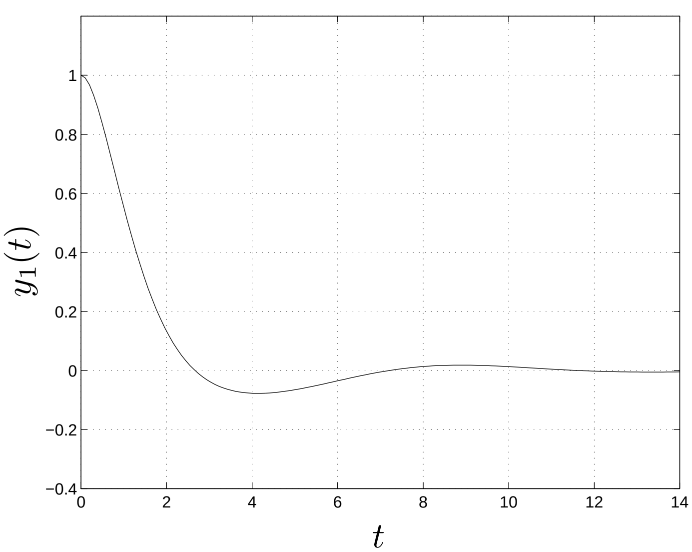
for $t_f = 3$, $u = u_{\rm ln}$ is:



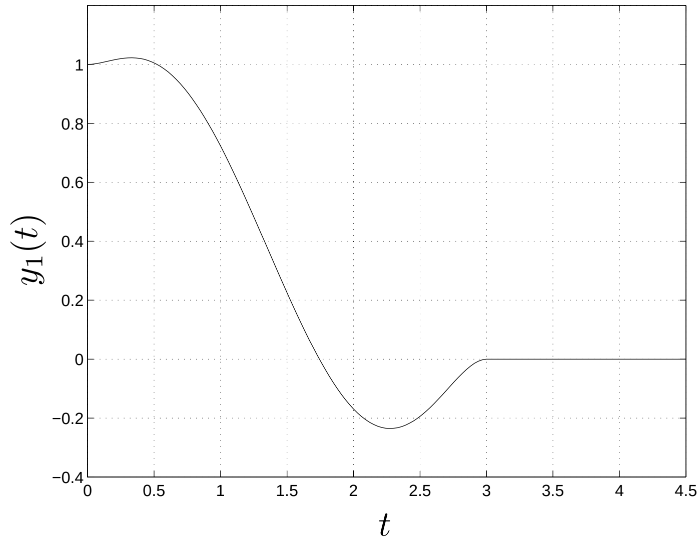
and for $t_f = 4$:



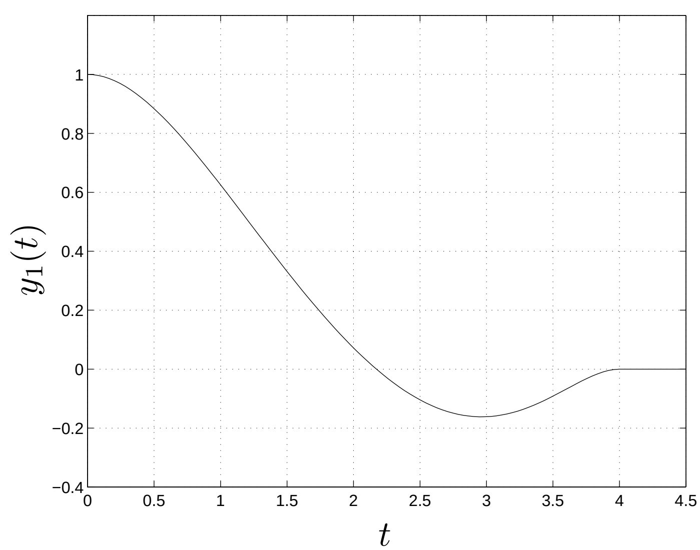
output y_1 for $u = 0$:



output y_1 for $u = u_{\text{ln}}$ with $t_f = 3$:



output y_1 for $u = u_{\text{ln}}$ with $t_f = 4$:



Lecture 19
Observability and state estimation

- state estimation
- discrete-time observability
- observability – controllability duality
- observers for noiseless case
- continuous-time observability
- least-squares observers
- example

State estimation set up

we consider the discrete-time system

$$x(t+1) = Ax(t) + Bu(t) + w(t), \quad y(t) = Cx(t) + Du(t) + v(t)$$

- w is state *disturbance* or *noise*
- v is sensor *noise* or *error*
- A, B, C, and D are known
- u and y are observed over time interval $[0, t-1]$
- w and v are not known, but can be described statistically, or assumed small ($e.g.$, in RMS value)

State estimation problem

state estimation problem: estimate $x(s)$ from

$$u(0), \ldots, u(t-1),\ y(0), \ldots, y(t-1)$$

- $s = 0$: estimate initial state
- $s = t-1$: estimate current state
- $s = t$: estimate ($i.e.$, predict) next state

an algorithm or system that yields an estimate $\hat{x}(s)$ is called an *observer* or *state estimator*

$\hat{x}(s)$ is denoted $\hat{x}(s|t-1)$ to show what information estimate is based on (read, "$\hat{x}(s)$ given $t-1$")

Noiseless case

let's look at finding $x(0)$, with no state or measurement noise:

$$x(t+1) = Ax(t) + Bu(t), \quad y(t) = Cx(t) + Du(t)$$

with $x(t) \in \mathbf{R}^n$, $u(t) \in \mathbf{R}^m$, $y(t) \in \mathbf{R}^p$

then we have

$$\begin{bmatrix} y(0) \\ \vdots \\ y(t-1) \end{bmatrix} = \mathcal{O}_t x(0) + \mathcal{T}_t \begin{bmatrix} u(0) \\ \vdots \\ u(t-1) \end{bmatrix}$$

where

$$\mathcal{O}_t = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{t-1} \end{bmatrix}, \quad \mathcal{T}_t = \begin{bmatrix} D & 0 & \cdots & & \\ CB & D & 0 & \cdots & \\ \vdots & & & & \\ CA^{t-2}B & CA^{t-3}B & \cdots & CB & D \end{bmatrix}$$

- $\mathcal{O}_t$ maps initials state into resulting output over $[0, t-1]$
- $\mathcal{T}_t$ maps input to output over $[0, t-1]$

hence we have

$$\mathcal{O}_t x(0) = \begin{bmatrix} y(0) \\ \vdots \\ y(t-1) \end{bmatrix} - \mathcal{T}_t \begin{bmatrix} u(0) \\ \vdots \\ u(t-1) \end{bmatrix}$$

RHS is known, $x(0)$ is to be determined

hence:

- can uniquely determine $x(0)$ if and only if $\mathcal{N}(\mathcal{O}_t) = \{0\}$
- $\mathcal{N}(\mathcal{O}_t)$ gives ambiguity in determining $x(0)$
- if $x(0) \in \mathcal{N}(\mathcal{O}_t)$ and $u = 0$, output is zero over interval $[0, t-1]$
- input u does not affect ability to determine $x(0)$;
  its effect can be subtracted out

Observability matrix

by C-H theorem, each A^k is linear combination of $A^0, \ldots, A^{n-1}$

hence for $t \geq n$, $\mathcal{N}(\mathcal{O}_t) = \mathcal{N}(\mathcal{O})$ where

$$\mathcal{O} = \mathcal{O}_n = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}$$

is called the *observability matrix*

if $x(0)$ can be deduced from u and y over $[0, t-1]$ for any t, then $x(0)$ can be deduced from u and y over $[0, n-1]$

$\mathcal{N}(\mathcal{O})$ is called *unobservable subspace*; describes ambiguity in determining state from input and output

system is called *observable* if $\mathcal{N}(\mathcal{O}) = \{0\}$, $i.e.$, $\mathbf{Rank}(\mathcal{O}) = n$

Observability – controllability duality

let $(\tilde{A}, \tilde{B}, \tilde{C}, \tilde{D})$ be dual of system (A, B, C, D), $i.e.$,

$$\tilde{A} = A^T, \quad \tilde{B} = C^T, \quad \tilde{C} = B^T, \quad \tilde{D} = D^T$$

controllability matrix of dual system is

$$\begin{aligned} \tilde{\mathcal{C}} &= [\tilde{B} \ \tilde{A}\tilde{B} \cdots \tilde{A}^{n-1}\tilde{B}] \\ &= [C^T \ A^T C^T \cdots (A^T)^{n-1} C^T] \\ &= \mathcal{O}^T, \end{aligned}$$

transpose of observability matrix

similarly we have $\tilde{\mathcal{O}} = \mathcal{C}^T$

thus, system is observable (controllable) if and only if dual system is controllable (observable)

in fact,

$$\mathcal{N}(\mathcal{O}) = \text{range}(\mathcal{O}^T)^\perp = \text{range}(\tilde{\mathcal{C}})^\perp$$

i.e., unobservable subspace is orthogonal complement of controllable subspace of dual

Observers for noiseless case

suppose $\mathbf{Rank}(\mathcal{O}_t) = n$ (*i.e.*, system is observable) and let F be any left inverse of $\mathcal{O}_t$, *i.e.*, $F\mathcal{O}_t = I$

then we have the observer

$$x(0) = F\left(\begin{bmatrix} y(0) \\ \vdots \\ y(t-1) \end{bmatrix} - \mathcal{T}_t \begin{bmatrix} u(0) \\ \vdots \\ u(t-1) \end{bmatrix}\right)$$

which deduces $x(0)$ (exactly) from u, y over $[0, t-1]$

in fact we have

$$x(\tau - t + 1) = F\left(\begin{bmatrix} y(\tau - t + 1) \\ \vdots \\ y(\tau) \end{bmatrix} - \mathcal{T}_t \begin{bmatrix} u(\tau - t + 1) \\ \vdots \\ u(\tau) \end{bmatrix}\right)$$

$i.e.$, our observer estimates what state was $t-1$ epochs ago, given past $t-1$ inputs & outputs

observer is (multi-input, multi-output) *finite impulse response* (FIR) filter, with inputs u and y, and output $\hat{x}$

Invariance of unobservable set

fact: the unobservable subspace $\mathcal{N}(\mathcal{O})$ is invariant, $i.e.$, if $z \in \mathcal{N}(\mathcal{O})$, then $Az \in \mathcal{N}(\mathcal{O})$

proof: suppose $z \in \mathcal{N}(\mathcal{O})$, $i.e.$, $CA^k z = 0$ for $k = 0, \ldots, n-1$

evidently $CA^k(Az) = 0$ for $k = 0, \ldots, n-2$;

$$CA^{n-1}(Az) = CA^n z = -\sum_{i=0}^{n-1} \alpha_i CA^i z = 0$$

(by C-H) where

$$\det(sI - A) = s^n + \alpha_{n-1}s^{n-1} + \cdots + \alpha_0$$

Continuous-time observability

continuous-time system with no sensor or state noise:

$$\dot{x} = Ax + Bu, \quad y = Cx + Du$$

can we deduce state x from u and y?

let's look at derivatives of y:

$$\begin{aligned} y &= Cx + Du \\ \dot{y} &= C\dot{x} + D\dot{u} = CAx + CBu + D\dot{u} \\ \ddot{y} &= CA^2x + CABu + CB\dot{u} + D\ddot{u} \end{aligned}$$

and so on

hence we have

$$\begin{bmatrix} y \\ \dot{y} \\ \vdots \\ y^{(n-1)} \end{bmatrix} = \mathcal{O}x + \mathcal{T} \begin{bmatrix} u \\ \dot{u} \\ \vdots \\ u^{(n-1)} \end{bmatrix}$$

where $\mathcal{O}$ is the observability matrix and

$$\mathcal{T} = \begin{bmatrix} D & 0 & \cdots & & \\ CB & D & 0 & \cdots & \\ \vdots & & & & \\ CA^{n-2}B & CA^{n-3}B & \cdots & CB & D \end{bmatrix}$$

(same matrices we encountered in discrete-time case!)

rewrite as

$$\mathcal{O}x = \begin{bmatrix} y \\ \dot{y} \\ \vdots \\ y^{(n-1)} \end{bmatrix} - \mathcal{T} \begin{bmatrix} u \\ \dot{u} \\ \vdots \\ u^{(n-1)} \end{bmatrix}$$

RHS is known; x is to be determined

hence if $\mathcal{N}(\mathcal{O}) = \{0\}$ we can deduce $x(t)$ from derivatives of $u(t)$, $y(t)$ up to order $n-1$

in this case we say system is observable

can construct an observer using any left inverse F of $\mathcal{O}$:

$$x = F\left(\begin{bmatrix} y \\ \dot{y} \\ \vdots \\ y^{(n-1)} \end{bmatrix} - \mathcal{T} \begin{bmatrix} u \\ \dot{u} \\ \vdots \\ u^{(n-1)} \end{bmatrix} \right)$$

- reconstructs $x(t)$ (exactly and instantaneously) from

$$u(t), \ldots, u^{(n-1)}(t),\ y(t), \ldots, y^{(n-1)}(t)$$

- derivative-based state reconstruction is dual of state transfer using impulsive inputs

A converse

suppose $z \in \mathcal{N}(\mathcal{O})$ (the unobservable subspace), and u is any input, with x, y the corresponding state and output, $i.e.$,

$$\dot{x} = Ax + Bu, \quad y = Cx + Du$$

then state trajectory $\tilde{x} = x + e^{tA}z$ satisfies

$$\dot{\tilde{x}} = A\tilde{x} + Bu, \quad y = C\tilde{x} + Du$$

$i.e.$, input/output signals u, y consistent with both state trajectories x, $\tilde{x}$

hence if system is unobservable, no signal processing of any kind applied to u and y can deduce x

unobservable subspace $\mathcal{N}(\mathcal{O})$ gives fundamental ambiguity in deducing x from u, y

Least-squares observers

discrete-time system, with sensor noise:

$$x(t+1) = Ax(t) + Bu(t), \quad y(t) = Cx(t) + Du(t) + v(t)$$

we assume $\mathbf{Rank}(\mathcal{O}_t) = n$ (hence, system is observable)

least-squares observer uses pseudo-inverse:

$$\hat{x}(0) = \mathcal{O}_t^\dagger \left(\begin{bmatrix} y(0) \\ \vdots \\ y(t-1) \end{bmatrix} - \mathcal{T}_t \begin{bmatrix} u(0) \\ \vdots \\ u(t-1) \end{bmatrix} \right)$$

where $\mathcal{O}_t^\dagger = \left(\mathcal{O}_t^T \mathcal{O}_t\right)^{-1} \mathcal{O}_t^T$

interpretation: $\hat{x}_{\rm ls}(0)$ minimizes discrepancy between

- output $\hat{y}$ that *would be* observed, with input u and initial state $x(0)$ (and no sensor noise), and
- output y that *was* observed,

measured as $\sum_{\tau=0}^{t-1} \|\hat{y}(\tau) - y(\tau)\|^2$

can express least-squares initial state estimate as

$$\hat{x}_{\rm ls}(0) = \left(\sum_{\tau=0}^{t-1} (A^T)^\tau C^T C A^\tau \right)^{-1} \sum_{\tau=0}^{t-1} (A^T)^\tau C^T \tilde{y}(\tau)$$

where $\tilde{y}$ is observed output with portion due to input subtracted: $\tilde{y} = y - h * u$ where h is impulse response

Least-squares observer uncertainty ellipsoid

since $\mathcal{O}_t^\dagger \mathcal{O}_t = I$, we have

$$\tilde{x}(0) = \hat{x}_{\rm ls}(0) - x(0) = \mathcal{O}_t^\dagger \begin{bmatrix} v(0) \\ \vdots \\ v(t-1) \end{bmatrix}$$

where $\tilde{x}(0)$ is the estimation error of the initial state

in particular, $\hat{x}_{\rm ls}(0) = x(0)$ if sensor noise is zero ($i.e.$, observer recovers exact state in noiseless case)

now assume sensor noise is unknown, but has RMS value $\leq \alpha$,

$$\frac{1}{t} \sum_{\tau=0}^{t-1} \|v(\tau)\|^2 \leq \alpha^2$$

set of possible estimation errors is ellipsoid

$$\tilde{x}(0) \in \mathcal{E}_{\rm unc} = \left\{ \mathcal{O}_t^\dagger \begin{bmatrix} v(0) \\ \vdots \\ v(t-1) \end{bmatrix} \;\middle|\; \frac{1}{t}\sum_{\tau=0}^{t-1} \|v(\tau)\|^2 \le \alpha^2 \right\}$$

$\mathcal{E}_{\rm unc}$ is 'uncertainty ellipsoid' for $x(0)$ (least-square gives best $\mathcal{E}_{\rm unc}$)

shape of uncertainty ellipsoid determined by matrix

$$\left(\mathcal{O}_t^T\mathcal{O}_t\right)^{-1} = \left(\sum_{\tau=0}^{t-1}(A^T)^\tau C^T C A^\tau\right)^{-1}$$

maximum norm of error is

$$\|\hat{x}_{\rm ls}(0) - x(0)\| \le \alpha\sqrt{t}\|\mathcal{O}_t^\dagger\|$$

Infinite horizon uncertainty ellipsoid

the matrix

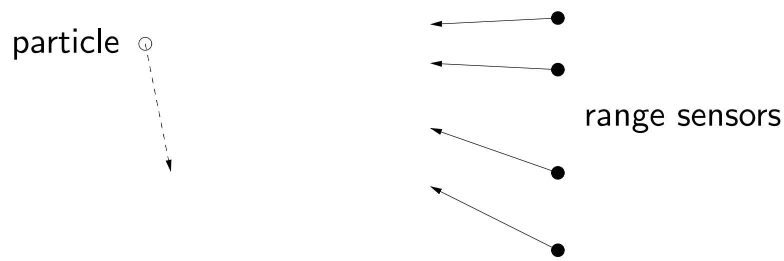
$$P = \lim_{t\to\infty}\left(\sum_{\tau=0}^{t-1}(A^T)^\tau C^T C A^\tau\right)^{-1}$$

always exists, and gives the limiting uncertainty in estimating $x(0)$ from u, y over longer and longer periods:

- if A is stable, $P > 0$
  $i.e.$, can't estimate initial state perfectly even with infinite number of measurements $u(t),\ y(t)$, $t = 0, \ldots$ (since memory of $x(0)$ fades . . .)
- if A is not stable, then P can have nonzero nullspace
  $i.e.$, initial state estimation error gets arbitrarily small (at least in some directions) as more and more of signals u and y are observed

Example

- particle in $\mathbf{R}^2$ moves with uniform velocity
- (linear, noisy) range measurements from directions $-15°$, $0°$, $20°$, $30°$, once per second
- range noises IID $\mathcal{N}(0,1)$; can assume RMS value of v is not much more than 2
- no assumptions about initial position & velocity



problem: estimate initial position & velocity from range measurements

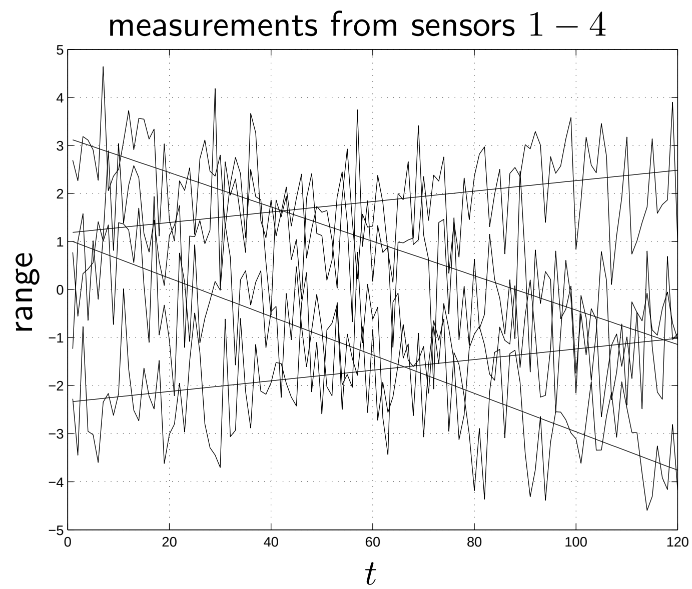
express as linear system

$$x(t+1) = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} x(t), \qquad y(t) = \begin{bmatrix} k_1^T \\ \vdots \\ k_4^T \end{bmatrix} x(t) + v(t)$$

- $(x_1(t), x_2(t))$ is position of particle
- $(x_3(t), x_4(t))$ is velocity of particle
- can assume RMS value of v is around 2
- k_i is unit vector from sensor i to origin

true initial position & velocities: $x(0) = (1\ \ -3\ \ -0.04\ 0.03)$

range measurements (& noiseless versions):

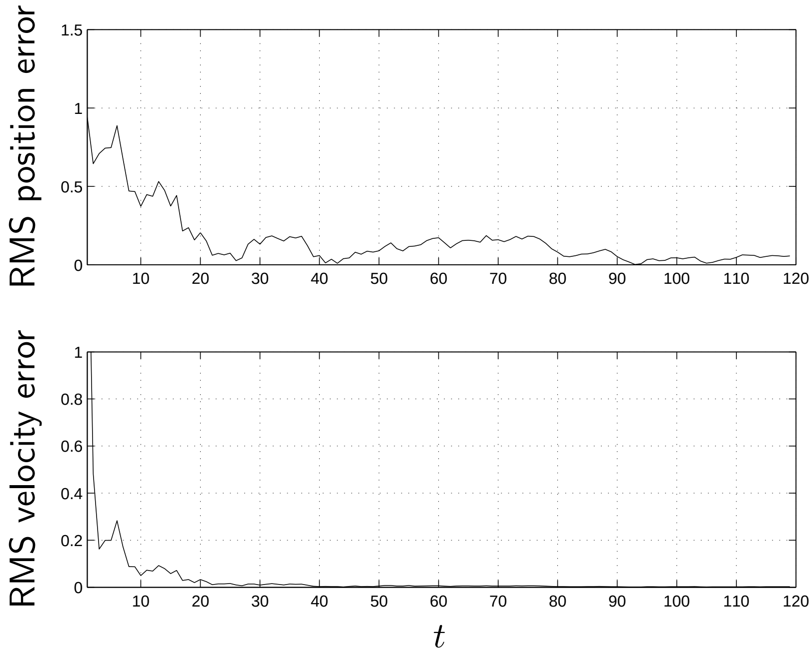


- estimate based on $(y(0), \ldots, y(t))$ is $\hat{x}(0|t)$

- actual RMS position error is

$$\sqrt{(\hat{x}_1(0|t) - x_1(0))^2 + (\hat{x}_2(0|t) - x_2(0))^2}$$

(similarly for actual RMS velocity error)



Continuous-time least-squares state estimation

assume $\dot{x} = Ax + Bu$, $y = Cx + Du + v$ is observable

least-squares estimate of initial state $x(0)$, given $u(\tau)$, $y(\tau)$, $0 \leq \tau \leq t$: choose $\hat{x}_{\rm ls}(0)$ to minimize integral square residual

$$J = \int_0^t \left\| \tilde{y}(\tau) - Ce^{\tau A}x(0) \right\|^2 \, d\tau$$

where $\tilde{y} = y - h * u$ is observed output minus part due to input

let's expand as $J = x(0)^T Q x(0) + 2r^T x(0) + s$,

$$Q = \int_0^t e^{\tau A^T} C^T C e^{\tau A} \, d\tau, \quad r = \int_0^t e^{\tau A^T} C^T \tilde{y}(\tau) \, d\tau,$$

$$q = \int_0^t \tilde{y}(\tau)^T \tilde{y}(\tau) \, d\tau$$

setting $\nabla_{x(0)} J$ to zero, we obtain the least-squares observer

$$\hat{x}_{\rm ls}(0) = Q^{-1} r = \left(\int_0^t e^{\tau A^T} C^T C e^{\tau A} \, d\tau \right)^{-1} \int_0^t e^{A^T \tau} C^T \tilde{y}(\tau) \, d\tau$$

estimation error is

$$\tilde{x}(0) = \hat{x}_{\rm ls}(0) - x(0) = \left(\int_0^t e^{\tau A^T} C^T C e^{\tau A} \, d\tau \right)^{-1} \int_0^t e^{\tau A^T} C^T v(\tau) \, d\tau$$

therefore if $v = 0$ then $\hat{x}_{\rm ls}(0) = x(0)$

Lecture 20
Some parting thoughts . . .

- linear algebra
- levels of understanding
- what's next?

Linear algebra

- comes up in *many* practical contexts (EE, ME, CE, AA, OR, Econ, . . .)
- nowadays is readily *done*
  cf. 10 yrs ago (when it was mostly *talked about*)

- Matlab or equiv for fooling around
- real codes ($e.g.$, LAPACK) widely available
- current level of linear algebra technology:
  - 500 – 1000 vbles: easy with general purpose codes
  - much more possible with special structure, special codes ($e.g.$, sparse, convolution, banded, . . .)

Levels of understanding

Simple, intuitive view:

- 17 vbles, 17 eqns: usually has unique solution
- 80 vbles, 60 eqns: 20 extra degrees of freedom

Platonic view:

- singular, rank, range, nullspace, Jordan form, controllability
- everything is precise & unambiguous
- gives insight & deeper understanding
- sometimes misleading in practice

Quantitative view:

- based on ideas like least-squares, SVD
- gives numerical measures for ideas like singularity, rank, etc.
- interpretation depends on (practical) context
- very useful in practice

- must have understanding at one level before moving to next
- **never forget** which level you are operating in

What's next?

- EE363 — linear dynamical systems (Win 08-09)
- EE364a — convex optimization I (Spr 08-09)

(plus lots of other EE, CS, ICME, MS&E, Stat, ME, AA courses on signal processing, control, graphics & vision, adaptive systems, machine learning, computational geometry, numerical linear algebra, . . .)