Linear Systems

Helwig Hauser *et al.*,
UiB Dept. of Informatics

Today

Fitness training with linearity:

- linear expressions
- linear combinations
- linear operators
- left- vs. right-multiplication (rows vs. columns)
- linear equations
- solving linear equations
  - Gauss elimination
  - LU decomposition
- singular matrices and non-square matrices
- using operators to solve linear equations

Introduction

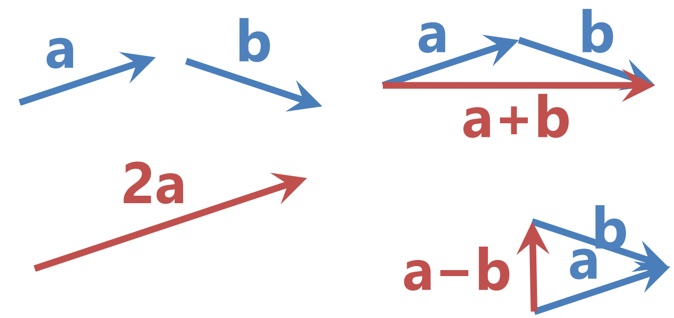
Linear system:

- system of linear equations (relations between variables)
- good to deal with, because
  - straight-forward to solve (if possible)
  - numerically stable (usually)
  - efficient
- lots *and lots* of real-world problems
  are mapped to / approximated by linear systems
- well supported by
  - tools like MATLAB, Mathematica, …
  - libraries like LAPACK, NumPy, …

Linearity in Math

Linear operations

- scalar multiplication: $\mathbf{c} = n\,\mathbf{a}$
- addition (& subtraction): $\mathbf{c} = \mathbf{a} + \mathbf{b}$
  ($\mathbf{d} = \mathbf{a} - \mathbf{b} = \mathbf{a} + -1\,\mathbf{b}$)



Expression "q is linear in x" (read also as "$q(x)$ is linear in x")

- interpretation:
  - if you vary x linearly, q (or $\mathbf{q}$) varies linearly (with x)
- q (or $\mathbf{q}$) is a function of x (maybe not only of x), for ex.:
  - $q(x) = q_{k,d}(x) = k\,x + d$ … is **linear in** x (k, d: parameters)
  - $q(x,y) = q_k(x,y) = y^2 + k\,x$ … is **linear in** x (not linear in y)
  - $\mathbf{q}(x) = (2 \;\; k\,x)^{\mathrm{T}}$ … is **linear in** x ($\mathbf{q}$: vector-valued function, 2D)

next slide…

Linearity in Math

Linear operations

– scalar multiplication: $\mathbf{c} = n\,\mathbf{a}$

– addition (& subtraction): $\mathbf{c} = \mathbf{a} + \mathbf{b}$
($\mathbf{d} = \mathbf{a} - \mathbf{b} = \mathbf{a} + -1\,\mathbf{b}$)

x	$q(x) = 2x + 1$
0	1
1	3
2	5
$\vdots$	$\vdots$

example

Expression "q is linear in x" (read also as "$q(x)$ is linear in x")

– interpretation:
  - if you vary x linearly, q (or $\mathbf{q}$) varies linearly with x

– q (or $\mathbf{q}$) is a function of x (maybe not only of x), for ex.:
  - $q(x) = q_{k,d}(x) = k\,x + d$ … is **linear in** x (k, d: parameters)
  - $q(x,y) = q_k(x,y) = y^2 + k\,x$ … is **linear in** x (not linear in y)
  - $\mathbf{q}(x) = (2 \;\; k\,x)^{\mathsf{T}}$ … is **linear in** x ($\mathbf{q}$: vector-valued function, 2D)

Rewriting Linear Expressions

The inner product: your friend! :–)

- often, it's useful to think of an expression $ax + by + \ldots$
  as of $\mathbf{u} \cdot \mathbf{v}$ with $\mathbf{u} = (a\ b\ \ldots)^T$ and $\mathbf{v} = (x\ y\ \ldots)^T$

- example:

  equation of a plane **p** in 3D: $3x + 4y + 5z = 12$

  becomes $(3\ 4\ 5) \cdot \mathbf{x} = 12$

  with $\mathbf{x} = (x\ \ y\ z)^T$ being an arbitrary point on plane **p**!

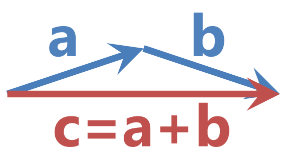
- even more useful: $\mathbf{n}^T \mathbf{x} = 12\ \ (= \mathbf{n}^T \mathbf{q})$
  with "parameters"
  - $\mathbf{n} = (3\ 4\ 5)^T$ being a normal of **p**
  - $\mathbf{q} = (1\ 1\ 1)^T$ being one point on **p**

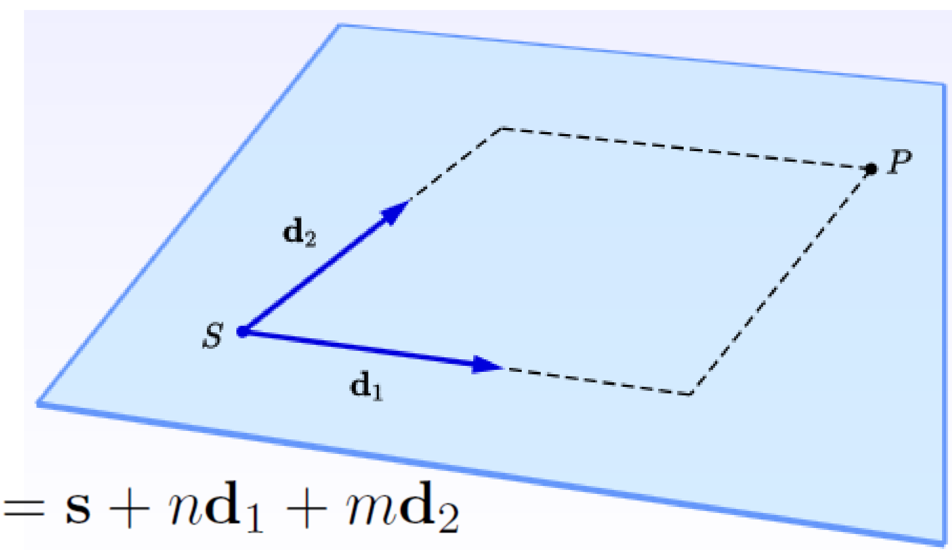
$\mathbf{x}$: variable of this equ.

Linear Combinations

Aka. weighted sum:

- **c** is a linear combination of **a** and **b** (and …),
  if **c** = *n* **a** + *m* **b** (+ …)
- for arbitrary *n* and *m*, *n* **a** + *m* **b** forms a plane through the origin:
  any **c** in that plane is a linear combination of **a** and **b**
- if **c** is a linear combination of **a** and **b** (and …),
  then **a**, **b**, and **c** (and …) are linearly dependent on each other





$$\mathbf{p} = \mathbf{p_s}(n, m) = \mathbf{p}(\mathbf{s}, n, m) = \mathbf{s} + n\mathbf{d}_1 + m\mathbf{d}_2$$

Linear Combination as an Operator

Operators:

- mapping from one vector space to another
  (more on vector spaces later)
- examples (we'll come back to them later, also):
  - operator ∇ maps to the vector of all partial derivatives
  - operator $\int_0^t$ maps to the antiderivative

Given some $\mathbf{x} = (x_1 \;\; x_2 \;\; \ldots \;\; x_n)^\mathsf{T}$ **and** $\mathbf{c} = (c_1 \;\; c_2 \;\; \ldots \;\; c_n)^\mathsf{T}$:

- $b = c_1 x_1 + c_2 x_2 + \ldots + c_n x_n$
  is a linear combination of all x_i
  with the weights / coefficients c_i
- using the inner product, we can write $b = \mathbf{c} \cdot \mathbf{x}$ or $<\mathbf{c}, \mathbf{x}>$
- using matrix multiplication, we can write $b = \mathbf{c}^\mathsf{T} \mathbf{x}$ or $\mathbf{c}^* \mathbf{x}$
- we can also write this linear combination as an operator $\Lambda_\mathbf{c}$,
  acting on $\mathbf{x}$, parameterized by $\mathbf{c}$: $b = \Lambda_\mathbf{c} \mathbf{x}$

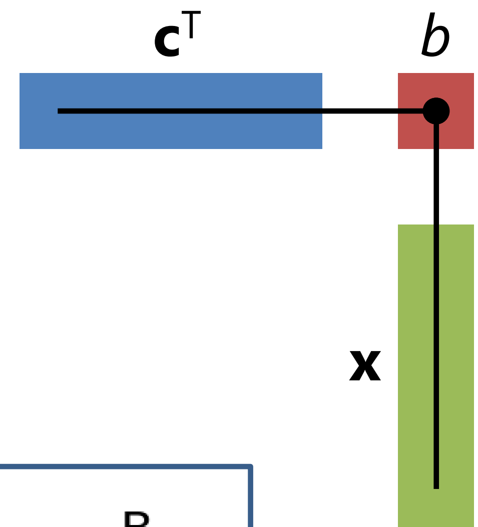
complex numbers

"Just" one Linear Combination

Given one vector x (of values x_i from $\mathbb{R}$)
and one vector of coefficients c:

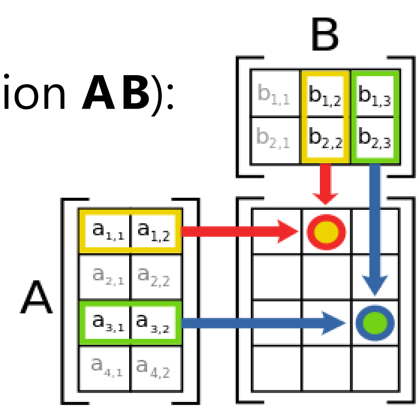
$$- \quad b = \mathbf{c}^\mathsf{T}\mathbf{x}$$

"read" right-to-left:
- starting with a vector $\mathbf{x}$,
- we use coefficients $\mathbf{c}$
- to compute b (*as* $\mathbf{c}^\mathsf{T}\mathbf{x}$)



$b \in \mathbb{R}$
$\mathbf{c} \in \mathbb{R}^n$
$\mathbf{x} \in \mathbb{R}^n$

Remember
(matrix multiplication **AB**):

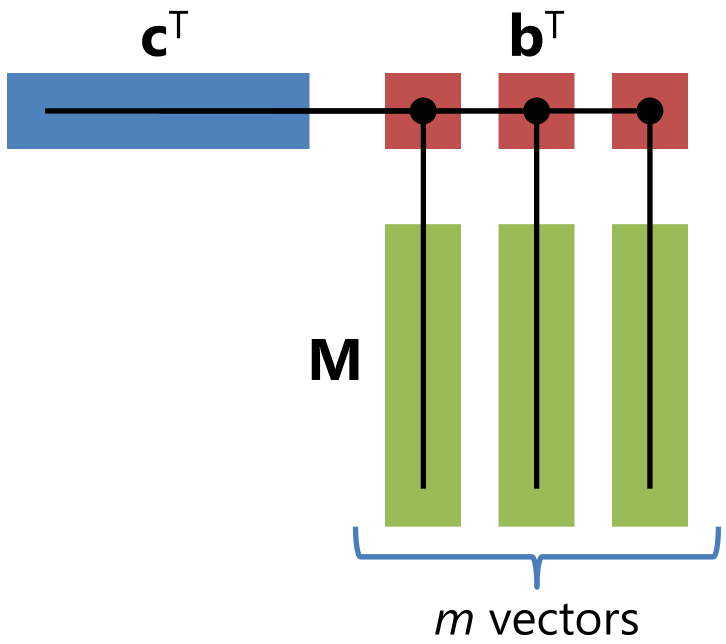


More 1a: a Linear Combination of m $\mathbf{x}_j$

Given m vectors $\mathbf{x}_j$
and one vector of coefficients $\mathbf{c}$:

– $\mathbf{b}^\mathsf{T} = \mathbf{c}^\mathsf{T}\,\mathbf{M}$

with $\mathbf{M} = (\,\mathbf{x}_1\ \mathbf{x}_2\ \ldots\ \mathbf{x}_m\,)$

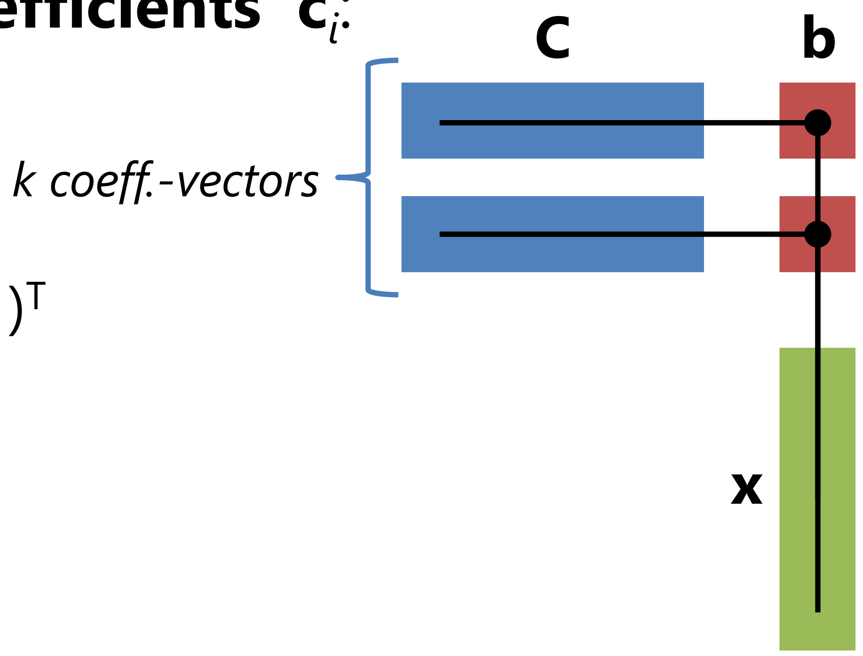


$\mathbf{b} \in \mathbb{R}^m$
$\mathbf{c} \in \mathbb{R}^n$
$\mathbf{M} \in \mathbb{R}^{n,m}$

After the multiplication,
$\mathbf{b}$ "hosts" the results of all the m linear combinations of $\mathbf{x}_j$
(all with the same vector of coefficients $\mathbf{c}$)

More 1b: k Linear Combs. "in parallel"

Given one vector x
and k vectors of coefficients $\mathbf{c}_i$:



$$-\ \mathbf{b} = \mathbf{C}\,\mathbf{x}$$

with $\mathbf{C} = (\mathbf{c}_1\ \mathbf{c}_2\ \ldots\ \mathbf{c}_k)^T$

$\mathbf{b} \in \mathbb{R}^k$
$\mathbf{C} \in \mathbb{R}^{k,n}$
$\mathbf{x} \in \mathbb{R}^n$

After the multiplication,
b "hosts" the results of k different linear combinations of **x**
(each one with its own vector of coefficients $\mathbf{c}_i$)

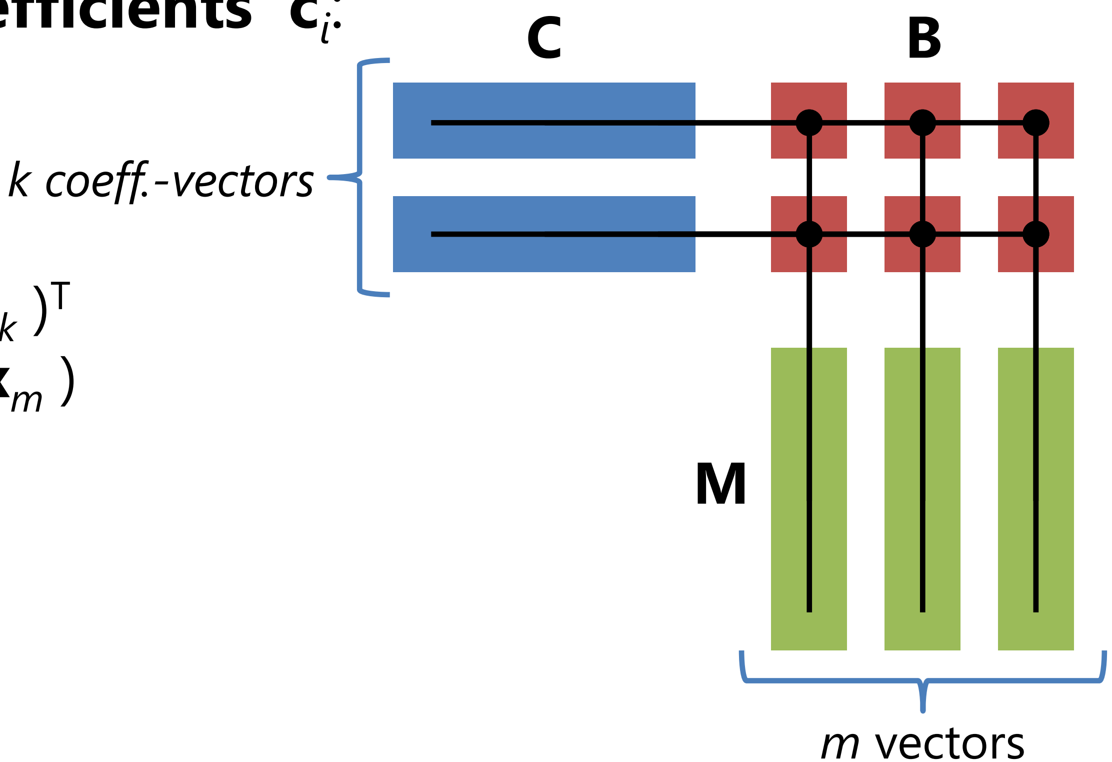
More 2: k LinCombs. of m $\mathbf{x}_j$

Given m **vectors** $\mathbf{x}_j$
and k **vectors of coefficients** $\mathbf{c}_i$:

– $\mathbf{B} = \mathbf{C}\,\mathbf{M}$

with $\mathbf{C} = (\mathbf{c}_1\ \mathbf{c}_2 \ldots \mathbf{c}_k)^{\mathrm{T}}$
with $\mathbf{M} = (\mathbf{x}_1\ \mathbf{x}_2 \ldots \mathbf{x}_m)$

$\mathbf{B} \in \mathbb{R}^{k,m}$
$\mathbf{C} \in \mathbb{R}^{k,n}$
$\mathbf{M} \in \mathbb{R}^{n,m}$



After the multiplication,
B “hosts” the results of k linear combinations of m vectors $\mathbf{x}_j$

Interpreting $\mathbf{D} = \mathbf{C}\,\mathbf{M} = \mathbf{\Lambda}\,\mathbf{M}$

Accordingly,
we can see the left-mulitiplication of an $n \times m$ matrix $\mathbf{M}$
with an $k \times n$ matrix $\mathbf{C}$

as applying a linear operation $\mathbf{\Lambda} = (\Lambda_1\ \Lambda_2\ \ldots\ \Lambda_k)^\top$ to $\mathbf{M}$,

resulting into an $k \times m$ matrix $\mathbf{D}$

with k rows $\mathbf{d}_i^\top = (d_{i,1}\ d_{i,2}\ \ldots\ d_{i,m})$,

each consisting of m components—
as many as there were (column-)vectors in $\mathbf{M}$.

Rows vs. Columns

The matrix product $\mathbf{A}\,\mathbf{x}$

- takes a $k \times n$ matrix $\mathbf{A}$ (can be "just" one row-vector – then $k=1$)
  and an $n \times m$ matrix $\mathbf{x}$ (can be "just" one column-vector – then $m=1$)
- and produces $\mathbf{b} = \mathbf{A}\,\mathbf{x}$,
  i.e., a $k \times m$ matrix $\mathbf{b}$ (k can be 1, and/or m can be 1)

$\mathbf{A}\,\mathbf{x}$ can be interpreted as

- the *left-multiplication* of $\mathbf{x}$ with $\mathbf{A}$ (**A** working on the ***rows*** of **x**)
  $\mathbf{b} = \mathbf{A}\,\mathbf{x}$
- the *right-multiplication* of $\mathbf{A}$ with $\mathbf{x}$ (**x** working on the ***columns*** of **A**)
  $\mathbf{A}\,\mathbf{x} = \mathbf{b}$

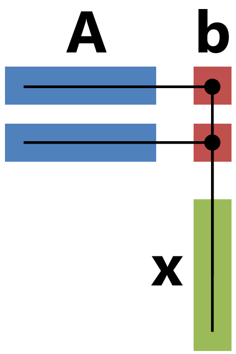
It's symmetric through transposition

- transposing $\mathbf{A}\,\mathbf{x} = \mathbf{b}$, i.e., forming $(\mathbf{A}\,\mathbf{x})^T = \mathbf{b}^T$
- leads to $\mathbf{b}^T = \mathbf{x}^T\,\mathbf{A}^T$ (rows became columns)

Left-Multiplication (**A** acting on **x**-rows)

Interpreting A x as *left-multiplying* x with A:

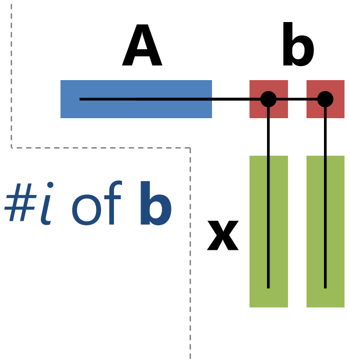
– every ***row*** of **b** = **A x**
is a linear combination of the rows of **x**
  - the coeffs. of the linear comb. producing row #*i* of **b** are in row #*i* of **A**



– examples:
  - **x** is an *n*-dimensional vector (*n* "rows")
  then **b** = **A x** is a *k*-dimensional vector
  where each of the *k* components of **b**
  is a linear combination of the *n* components of **x**
  (with the rows of **A** hosting *n* coeffs. per combination)
  - **x** is an *n* x *m* matrix
  and **b** = **A x** can be an elimination operation on **x** (more on elim. later), for ex., to eliminate $x_{2,1}$, $\mathbf{E}_{2,1}$ would be chosen as **A**

Right-Multiplication (**x** acting on **A**-cols.)

Interpreting A x as *right-multiplying* A with x:



– every ***column*** of **b** = **A x**
is a linear combination of the columns of **A**
  - the coeffs. for the linear comb. producing column #*i* of **b** are in column #*i* of **x**

– examples:
  - **x** is an *n*-dimensional vector (*n* "rows", here coefficients) and **b** = **A x** is a *k*-dimensional vector which is a linear combination of the *n* columns of **A** (with the elements of **x** being the *n* coeffs.)
  - given **A** as a *k* x *n* matrix;
then, for all possible $\mathbf{x} \in \mathbb{R}^n$, **A x** is a sub-space of $\mathbb{R}^k$,
i.e., the image of $\mathbb{R}^n$ under mapping $\mathbf{M}(\mathbf{x}) = \mathbf{A}\,\mathbf{x}$

Transposition Switches Roles

Given m **vectors** $\mathbf{x}_j$
and k **vectors of coefficients** $\mathbf{c}_i$:

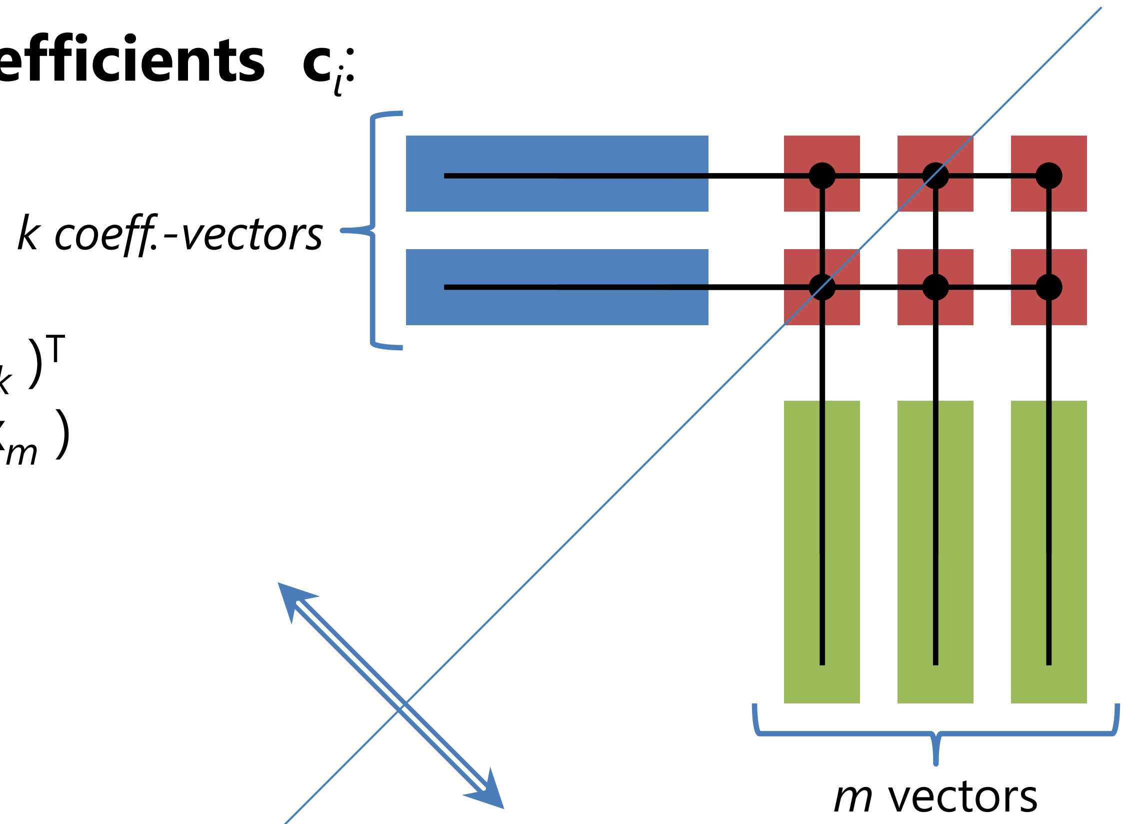
- $\mathbf{B} = \mathbf{C}\,\mathbf{M}$

  with $\mathbf{C} = (\,\mathbf{c}_1\ \mathbf{c}_2\ \dots\ \mathbf{c}_k\,)^{\mathsf{T}}$
  with $\mathbf{M} = (\,\mathbf{x}_1\ \mathbf{x}_2\ \dots\ \mathbf{x}_m\,)$

  $\mathbf{B} \in \mathbb{R}^{k,m}$
  $\mathbf{C} \in \mathbb{R}^{k,n}$
  $\mathbf{M} \in \mathbb{R}^{n,m}$

- after transposition:

  $\mathbf{M}^{\mathsf{T}}\,\mathbf{C}^{\mathsf{T}} = \mathbf{B}^{\mathsf{T}}$



transposition
flips the roles of **C** and **M**
and their rows and columns

Linear Equations

A linear equation is

- an algebraic equation in which each term
  - is either a constant (like parameter d or number 2)
  - or the product of a constant and a single variable (like $k\,x$)

    $\rightarrow$ no products of variables! $\rightarrow$ parameters are not variables!
- constants may be numbers, parameters, linear or even non-linear functions of parameters (but since parameters don't change, also their functions do not)

Linear Equations, examples

A linear equation with one unknown x (1D scenario)

- can be written as $ax = b$ with one solution $x = b/a$ (if $a \neq 0$)

Forms of linear equations with two unknowns x **and** y **:**

- standard form: $ax + by = c$
- slope-intercept form: $y = mx + b$ (no "vertical" lines!)
- intercept form: $x/a + y/b = 1$ … x-crossing at x=a, y-cr. at y=b
- point-slope form: $y - y_1 = m(x - x_1)$ … Δy proportional to Δx
- matrix form: $\mathbf{a}^\mathrm{T}\mathbf{x} = c$ (or $\mathbf{a}^*\mathbf{x} = c$ in the case of complex numbers)
- …

ChemBob Example

Bob buys supply for his chemistry lab—he buys

- small containers of lead nitrate (n), sulfur (s), and magnesium chloride (c)
- the store offers the following (per container):
  - lead nitrate: 90g (↔ 20cm^3) for \$14,25
  - sulfur: 40g (↔ 20cm^3) for \$5
  - magn. chloride: 30g (↔ 20cm^3) for \$2,75
- Bob buys in total 800g (↔ 340cm^3) and pays \$104

? How many containers of lead nitrate, sulfur, and magnesium chloride did he purchase?

n:


s:


c:


ChemBob, (manual) **solution**

Bob buys n containers with lead nitrate, s with sulfur, and c containers with magnesium chloride:

$$\begin{aligned}
n \cdot 90\text{g} \quad &+ s \cdot 40\text{g} &+ c \cdot 30\text{g} \quad &= 800\text{g} \qquad &(1.)\\
n \cdot 20\text{cm}^3 &+ s \cdot 20\text{cm}^3 &+ c \cdot 20\text{cm}^3 &= 340\text{cm}^3 &(2.)\\
n \cdot 14.25\$ &+ s \cdot 5\$ &+ c \cdot 2.75\$ &= 104\$ &(3.)
\end{aligned}$$

$$\begin{aligned}
9n + 4s + 3c &= 80 \qquad &(\alpha\text{: } 1./10\text{g})\\
n + s + c &= 17 &(\beta\text{: } 2./20\text{cm}^3) \qquad \Leftarrow\\
57n + 20s + 11c &= 416 &(\gamma\text{: } 3. \cdot 4/\$)
\end{aligned}$$

$$\begin{aligned}
5s + 6c &= 73 \qquad &(\text{I: } 9\beta - \alpha) \qquad \Leftarrow\\
37s + 46c &= 553 &(\text{II: } 57\beta - \gamma)
\end{aligned}$$

$$4s = 20 \qquad (\Sigma\text{: } 23\,\text{I} - 3\,\text{II}) \qquad \Leftarrow$$

$$\boxed{s = 5}\,(\Sigma), \boxed{c = 8}\,(\text{I: } 25 + 6c = 73), \boxed{n = 4}\,(\beta\text{: } n + 5 + 8 = 17)$$

ChemBob, (manual) **solution**

Bob buys n containers with lead nitrate, s with sulfur, and c containers with magnesium chloride:

$$
\begin{array}{lllll}
n \cdot 90\text{g} & + s \cdot 40\text{g} & + c \cdot 30\text{g} & = 800\text{g} & (1.) \\
n \cdot 20\text{cm}^3 & + s \cdot 20\text{cm}^3 & + c \cdot 20\text{cm}^3 & = 340\text{cm}^3 & (2.) \\
n \cdot 14.25\$ & + s \cdot 5\$ & + c \cdot 2.75\$ & = 104\$ & (3.)
\end{array}
$$

messy

$$
\begin{array}{rcll}
9n + 4s + 3c & = & 80 & (\alpha\text{: } 1./10\text{g}) \\
n + s + c & = & 17 & (\beta\text{: } 2./20\text{cm}^3) \quad \Leftarrow \\
57n + 20s + 11c & = & 416 & (\gamma\text{: } 3./0.25\$) \\
 & & & \\
5s + 6c & = & 73 & (\text{I: } 9\beta - \alpha) \quad \Leftarrow \\
37s + 46c & = & 553 & (\text{II: } 57\beta - \gamma) \\
 & & & \\
4s & = & 20 & (\Sigma\text{: } 23\,\text{I} - 3\,\text{II}) \quad \Leftarrow
\end{array}
$$

error-prone

☹

$\boxed{s = 5}\,(\Sigma)$, $\boxed{c = 8}\,(\text{I: } 25 + 6c = 73)$, $\boxed{n = 4}\,(\beta\text{: } n + 5 + 8 = 17)$

ChemBob, (manual) solution – what?

Bob buys n containers with lead nitrate, s with sulfur, and c containers with magnesium chloride:

$$
\begin{aligned}
n \cdot 90\text{g} \quad &+ s \cdot 40\text{g} \quad + c \cdot 30\text{g} &= 800\text{g} \quad && (1.)\\
n \cdot 20\text{cm}^3 &+ s \cdot 20\text{cm}^3 + c \cdot 20\text{cm}^3 &= 340\text{cm}^3 && (2.)\\
n \cdot 14.25\$ &+ s \cdot 5\$ \quad + c \cdot 2.75\$ &= 104\$ && (3.)
\end{aligned}
$$

Ax=b

$$
\boxed{
\begin{aligned}
9n + 4s + 3c &= 80\\
n + s + c &= 17\\
57n + 20s + 11c &= 416
\end{aligned}
}
$$

Gauss elimination

$(\beta: 2./20\text{cm}^3) \Leftarrow$

$$
\begin{aligned}
5s + 6c &= 73 \quad && (\mathrm{I}: 9\beta - \alpha) \quad \Leftarrow\\
37s + 46c &= 553 && (\mathrm{II}: 57\beta - \gamma)
\end{aligned}
$$

$(\Sigma: 23\,\mathrm{I} - 3\,\mathrm{II}) \Leftarrow$

back-substitution

$\boxed{s = 5}\,(\Sigma)$, $\boxed{c = 8}\,(\mathrm{I}:\ 25 + 6c = 73)$, $\boxed{n = 4}\,(\beta:\ n + 5 + 8 = 17)$

ChemBob, linear algebra setup

Input: masses, volumes, prices, totals (without units)

- $\mathbf{m} = (90, 40, 30)^\top$; $\mathbf{v} = (20, 20, 20)^\top$; $\mathbf{p} = (14.25, 5, 2.75)^\top$
- $\mathbf{t} = (t_\mathrm{m}, t_\mathrm{v}, t_\mathrm{p})^\top = (800, 340, 104)^\top$
- $\mathbf{x} = (n, s, c)^\top$ – the yet unknown numbers of containers

Setup: inner products $\Rightarrow$ matrix multiplication

- $\boxed{\mathbf{m}^\top\mathbf{x}} = 90\,n + 40\,s + 30\,c \;\boxed{= t_\mathrm{m}} = 800;$
- $\boxed{\mathbf{v}^\top\mathbf{x}} = 20\,n + 20\,s + 20\,c \;\boxed{= t_\mathrm{v}} = 340;$
- $\boxed{\mathbf{p}^\top\mathbf{x}} = 14.25\,n + 5\,s + 2.75\,c \;\boxed{= t_\mathrm{p}} = 104$

$$\mathbf{A}\mathbf{x} = \mathbf{t} \quad \text{with} \quad \mathbf{A} = \begin{pmatrix} \mathbf{m}^\top \\ \mathbf{v}^\top \\ \mathbf{p}^\top \end{pmatrix}$$

$$\Rightarrow \qquad \begin{pmatrix} 90 & 40 & 30 \\ 20 & 20 & 20 \\ 14.25 & 5 & 2.75 \end{pmatrix} \begin{pmatrix} n \\ s \\ c \end{pmatrix} = \begin{pmatrix} 800 \\ 340 \\ 104 \end{pmatrix}$$

ChemBob, linear algebra setup

Input: masses, volumes, prices, totals (without units)

- $\mathbf{m} = (90, 40, 30)^\mathsf{T}$; $\mathbf{v} = (20, 20, 20)^\mathsf{T}$; $\mathbf{p} = (14.25, 5, 2.75)^\mathsf{T}$
- $\mathbf{t} = (t_\mathrm{m}, t_\mathrm{v}, t_\mathrm{p})^\mathsf{T} = (800, 340, 104)^\mathsf{T}$
- $\mathbf{x} = (n, s, c)^\mathsf{T}$ – the yet unknown numbers of containers

Setup: inner products $\Rightarrow$ matrix multiplication

$$\left.\begin{aligned} \mathbf{m}^\mathsf{T}\mathbf{x} &= 90\,n + 40\,s + 30\,c &&= t_\mathrm{m} = 800;\\ \mathbf{v}^\mathsf{T}\mathbf{x} &= 20\,n + 20\,s + 20\,c &&= t_\mathrm{v} = 340;\\ \mathbf{p}^\mathsf{T}\mathbf{x} &= 14.25\,n + 5\,s + 2.75\,c &&= t_\mathrm{p} = 104 \end{aligned}\right\} \quad \mathbf{A}\mathbf{x} = \mathbf{t} \quad \text{with} \quad \mathbf{A} = \begin{pmatrix} \mathbf{m}^\top \\ \mathbf{v}^\top \\ \mathbf{p}^\top \end{pmatrix}$$

nice!

$$\Rightarrow \begin{pmatrix} 90 & 40 & 30 \\ 20 & 20 & 20 \\ 14.25 & 5 & 2.75 \end{pmatrix} \begin{pmatrix} n \\ s \\ c \end{pmatrix} = \begin{pmatrix} 800 \\ 340 \\ 104 \end{pmatrix}$$

ChemBob, MATLAB solution

[see MATLAB example 2a_A]

in MATLAB: „`m = [90; 40; 30]; v = [20; 20; 20]; p = [14.25; 5; 2.75]; A = [m.'; v.'; p.']; t = [800; 340; 104];`“ and then „`x = A\t`“

Setup: inner products $\Rightarrow$ matrix multiplication

$$\left.\begin{aligned}
&- \quad \mathbf{m}^\top \mathbf{x} = 90\,n + 40\,s + 30\,c = t_\mathrm{m} = 800;\\
&- \quad \mathbf{v}^\top \mathbf{x} = 20\,n + 20\,s + 20\,c = t_\mathrm{v} = 340;\\
&- \quad \mathbf{p}^\top \mathbf{x} = 14.25\,n + 5\,s + 2.75\,c = t_\mathrm{p} = 104
\end{aligned}\right\} \mathbf{A}\mathbf{x} = \mathbf{t} \quad \text{with} \quad \mathbf{A} = \begin{pmatrix} \mathbf{m}^\top \\ \mathbf{v}^\top \\ \mathbf{p}^\top \end{pmatrix}$$

$$\Rightarrow \quad \begin{pmatrix} 90 & 40 & 30 \\ 20 & 20 & 20 \\ 14.25 & 5 & 2.75 \end{pmatrix} \begin{pmatrix} n \\ s \\ c \end{pmatrix} = \begin{pmatrix} 800 \\ 340 \\ 104 \end{pmatrix}$$

Gaussian Elimination

Idea: eliminate *n*-1 **variables, one by one**

– bring the equation system into triangular form (this allows to solve it by back-substitution!)

$$\begin{aligned} 6x + y - 4z &= 4 \\ 2y + 3z &= -5 \\ 11z &= -33 \end{aligned}$$

Useful operations not affecting the solution:

– swapping equations (rows)

$$\left.\begin{aligned} x + y &= 2 \\ x - y &= 0 \end{aligned}\right\} = \left\{\begin{aligned} x - y &= 0 \\ x + y &= 2 \end{aligned}\right.$$

– multiplying equations (rows) by a (non-zero) scalar

$$\left.x + y = 2\right\} = \left\{2x + 2y = 4\right.$$

– adding equations (rows) to each other

$$\left.\begin{aligned} x + y &= 2 \\ x - y &= 0 \end{aligned}\right\} = \left\{\begin{aligned} x + y &= 2 \\ 2x &= 2 \end{aligned}\right.$$

Gaussian Elimination

Augmented matrix

– since we operate on both sides of the equations,
we form the augmented matrix: **M** = (**A** | **b**)

$$\begin{pmatrix} 90 & 40 & 30 \\ 20 & 20 & 20 \\ 14.25 & 5 & 2.75 \end{pmatrix} \begin{pmatrix} n \\ s \\ c \end{pmatrix} = \begin{pmatrix} 800 \\ 340 \\ 104 \end{pmatrix} \quad \Rightarrow \quad \left(\begin{array}{ccc|c} 90 & 40 & 30 & 800 \\ 20 & 20 & 20 & 340 \\ 14.25 & 5 & 2.75 & 104 \end{array} \right)$$

$\rightarrow$ this way, we keep track of both sides!

Steps:

1. choose a non-zero pivot element from the left-most column (one with a large absolute value is usually a good choice)
2. form new rows by "intelligent" multiplications and additions so that the matrix-elements in the pivot-column become 0
3. proceed to the next column and repeat (the pivot-row from above stays as it was)

Gaussian Elimination, ChemBob ex.

Choosing 90 as the pivot-element in $(a_{i,1})$:

$$\left(\begin{array}{ccc|c} \underline{90} & 40 & 30 & 800 \\ 20 & 20 & 20 & 340 \\ 14.25 & 5 & 2.75 & 104 \end{array}\right)$$

To "eliminate" $a_{2,1}$: new $(a_{2,j}) = 20(a_{1,j}) - 90(a_{2,j})$,
to "eliminate" $a_{3,1}$: new $(a_{3,j}) = 14.25(a_{1,j}) - 90(a_{3,j})$:

$$\left(\begin{array}{ccc|c} 90 & 40 & 30 & 800 \\ \underline{0} & -1000 & -1200 & -14\,600 \\ \underline{0} & 120 & 180 & 2040 \end{array}\right)$$

Gaussian Elimination, ChemBob ex.

Dividing the (new) 2. row by -100, the (new) 3. row by 10, and choosing 12 as the new pivot-element, we get:

$$\left(\begin{array}{ccc|c} 90 & 40 & 30 & 800 \\ 0 & \underline{12} & 18 & 204 \\ 0 & 10 & 12 & 146 \end{array}\right)$$

To "eliminate" $\bar{a}_{3,2}$: new $(\bar{a}_{3,j}) = 10(\bar{a}_{2,j}) - 12(\bar{a}_{3,j})$:

$$\left(\begin{array}{ccc|c} 90 & 40 & 30 & 800 \\ \underline{0} & 12 & 18 & 204 \\ \underline{0} & \underline{0} & 36 & 288 \end{array}\right)$$

Gaussian Elimination, ChemBob ex.

Thus, system

$$\left(\begin{array}{ccc|c} 90 & 40 & 30 & 800 \\ 20 & 20 & 20 & 340 \\ 14.25 & 5 & 2.75 & 104 \end{array}\right)$$

became

$$\left(\begin{array}{ccc|c} 90 & 40 & 30 & 800 \\ \underline{0} & 12 & 18 & 204 \\ \underline{0} & \underline{0} & 36 & 288 \end{array}\right)$$

note the change on the right side!

Back-substitution leads to:

$$\begin{aligned} 36\,c &= 288 &\Rightarrow\quad c &= 8 \\ 12\,s + 18 \cdot 8 &= 204 &\Rightarrow\quad s &= 5 \\ 90\,n + 40 \cdot 5 + 30 \cdot 8 &= 800 &\Rightarrow\quad n &= 4 \end{aligned}$$

Gauss Elimination as Operation

One elimination operation = left-multiplication with an elimination matrix

– If $\mathcal{E}_{2,1}$ is the operation to eliminate element $a_{2,1}$ from $\mathbf{A}$, then $\mathcal{E}_{2,1}$ corresponds to left-multiplying $\mathbf{A}$ with $\mathbf{E}_{2,1}$:

$$\mathbf{E}_{2,1} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ -\frac{a_{2,1}}{a_{1,1}} & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{pmatrix}$$

a certain multitude of the pivot row (#1) is subtracted from **A**'s row #2

all rows of **A** (except for row #2) stay the same (for now)

Left-multiplying $\mathbf{A}$ with $\mathbf{E}_{2,1}$ results into a new matrix $\mathbf{A}'$ that is exactly like $\mathbf{A}$ only that its 2nd row $(a'_{2,j})$ is a linear combination of rows 1 & 2 of $\mathbf{A}$: $(a'_{2,j}) = (a_{2,j}) - \frac{a_{2,1}}{a_{1,1}}(a_{1,j})$ with $a'_{2,1} = a_{2,1} - \frac{a_{2,1}}{a_{1,1}} a_{1,1} = 0$ (as intended).

$\mathbf{A}\mathbf{x} = \mathbf{b}$ transforms then into $\mathbf{A}'\mathbf{x} = \mathbf{b}'$, accordingly, with $\mathbf{A}' = \mathbf{E}_{2,1}\mathbf{A}$ and $\mathbf{b}' = \mathbf{E}_{2,1}\mathbf{b}$ ($\mathbf{b}'$ is the same as $\mathbf{b}$ with the exception that $b'_2 = b_2 - \frac{a_{2,1}}{a_{1,1}} b_1$).

left-multiplication: working on **A**-rows: $\mathbf{A}' = \mathbf{E}_{2,1}\mathbf{A}$

Interpreting Elimination Operations

Matrix to eliminate $a_{i,j}$ **from A** ($i>j$):

$$\mathbf{E}_{i,j} = \left(\begin{array}{ccccccc}
1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\
\vdots & \ddots & \vdots & & \vdots & & \vdots \\
0 & \cdots & 1 & \cdots & 0 & \cdots & 0 \\
\vdots & & \vdots & \ddots & \vdots & & \vdots \\
0 & \cdots & \underline{-\frac{a_{i,j}}{a_{j,j}}} & \cdots & \underline{1} & \cdots & 0 \\
\vdots & & \vdots & & \vdots & \ddots & \vdots \\
0 & \cdots & 0 & \cdots & 0 & \cdots & 1
\end{array}\right)
\begin{array}{l}
\\
\\
\text{... pivot-row \#}j \\
\\
\text{... row \#}i \\
\\
\\
\end{array}$$

Interpretation:

– new row #i = old row #i minus $\frac{a_{i,j}}{a_{j,j}}$ times the pivot row (#j)

note: the pivot row (#j) of **A** has $a_{j,j}$ in the j-th column

Eliminating An Entire Column

To eliminate all $a_{i,j}$ **with** $i>j$ **from A:**

$$
\mathbf{E}_j = \begin{pmatrix}
1 & \cdots & 0 & 0 & \cdots & 0 \\
\vdots & \ddots & \vdots & \vdots & & \vdots \\
0 & \cdots & 1 & 0 & \cdots & 0 \\
0 & \cdots & -\frac{a_{j+1,j}}{a_{j,j}} & 1 & \cdots & 0 \\
\vdots & & \vdots & \vdots & \ddots & \vdots \\
0 & \cdots & -\frac{a_{n,j}}{a_{j,j}} & 0 & \cdots & 1
\end{pmatrix}
$$

... row #*j*

... use the pivot row (#*j*)
to adjust all rows #*i* > #*j*
by elimination

Interpretation:

– subtract the appropriate multitude of the pivot row (#*j*) from all rows underneath (#$i>j$) in order to eliminate all $a_{i,j}$ with $i>j$

Gaussian Elimination as Operation(s)

We can now write the Gaussian elimination as operation(s):

– $$\mathbf{U}\,\mathbf{x} = \mathbf{E}\,\mathbf{A}\,\mathbf{x} = \underbrace{\mathbf{E}_{n-1}\,\mathbf{E}_{n-2}\cdots\mathbf{E}_1}_{\mathbf{E}}\,\mathbf{A}\,\mathbf{x} =$$
$$= \underbrace{\mathbf{E}_{n-1}\,\mathbf{E}_{n-2}\cdots\mathbf{E}_1}_{\mathbf{E}}\,\mathbf{b} = \mathbf{E}\,\mathbf{b} = \mathbf{d}$$

note the change
on the right side!

ChemBob Example, another day

Bob buys again for his chemistry lab—he buys

- small containers of lead nitrate (n), sulfur (s), and magnesium chloride (c)
- the store offers the following (per container):
  - lead nitrate: 90g (↔ 20cm^3) for $14,25
  - sulfur: 40g (↔ 20cm^3) for $5
  - magn. chloride: 30g (↔ 20cm^3) for $2,75
- Bob now buys in total 640g (↔ 300cm^3) and pays $80

? How many containers of lead nitrate, sulfur, and magnesium chloride did he purchase?

n: 

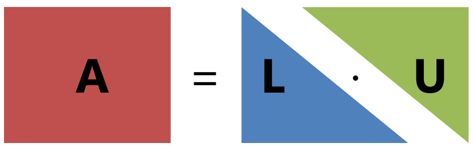
s: 

c: 

What now? All Gaussian elimination—*O*(n^3)!—again?

LU Decomposition

Rewriting A into product L U …



- **L**: lower-diagonal matrix
- **U**: upper-diagonal matrix

… makes repeated solving cheaper

- system $\mathbf{A}\,\mathbf{x} = \mathbf{b}$ becomes $\mathbf{L}\,\mathbf{U}\,\mathbf{x} = \mathbf{b}$ can be split into two systems
  - by setting $\mathbf{U}\,\mathbf{x} = \mathbf{y}$
  - we get $\mathbf{L}\,\mathbf{y} = \mathbf{b}$ → no change on the right-hand side!

compare to
$\mathbf{U}\,\mathbf{x} = \mathbf{E}\,\mathbf{A}\,\mathbf{x} = \mathbf{E}\,\mathbf{b} = \mathbf{d}$
in Gauss elimination

- both systems can be solved by substitution ($O(n^2)$):
  1. solve $\mathbf{L}\,\mathbf{y} = \mathbf{b}$ for $\mathbf{y}$ by forward-substitution
  2. solve $\mathbf{U}\,\mathbf{x} = \mathbf{y}$ for $\mathbf{x}$ by back-substitution

LU Decomposition, how-to

Nice case first (**A non-singular**, i.e., **invertible**):

– we start from **A** = **I A** (with **I** being the identity matrix), example (from the book):

$$\mathbf{A} = \begin{pmatrix} 4 & 3 & -1 \\ -2 & -4 & 5 \\ 1 & 2 & 6 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 4 & 3 & -1 \\ -2 & -4 & 5 \\ 1 & 2 & 6 \end{pmatrix}$$

– similar to Gaussian elimination, we
  - transform **A** into **U** (row by row)
  - and **I** into **L** (simultaneously)

LU Decomposition, example

$$\mathbf{A} = \begin{pmatrix} 4 & 3 & -1 \\ -2 & -4 & 5 \\ 1 & 2 & 6 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 4 & 3 & -1 \\ -2 & -4 & 5 \\ 1 & 2 & 6 \end{pmatrix}$$

In Gaussian elimination, we "eliminated" $a_{2,1} = -2$ by forming a new row $(\bar{a}_{2,j}) = (a_{2,j}) + \frac{1}{2}(a_{1,j})$:

$$(\bar{a}_{2,j}) = \begin{pmatrix} 0 & -2.5 & 4.5 \end{pmatrix}$$

Putting $(a_{2,j})$ to the left (as above!),
we get $(a_{2,j}) = -\frac{1}{2}(a_{1,j}) + (\bar{a}_{2,j})$ and thus

$$\mathbf{A} = \begin{pmatrix} 4 & 3 & -1 \\ -2 & -4 & 5 \\ 1 & 2 & 6 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ -\frac{1}{2} & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 4 & 3 & -1 \\ 0 & -2.5 & 4.5 \\ 1 & 2 & 6 \end{pmatrix}$$

LU Decomposition, example

$$\mathbf{A} = \begin{pmatrix} 4 & 3 & -1 \\ -2 & -4 & 5 \\ 1 & 2 & 6 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ -\frac{1}{2} & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 4 & 3 & -1 \\ 0 & -2.5 & 4.5 \\ 1 & 2 & 6 \end{pmatrix}$$

Accordingly, we "eliminate" $a_{3,1} = 1$
by forming $(\bar{a}_{3,j}) = (a_{3,j}) - \frac{1}{4}(a_{1,j})$:

$$(\bar{a}_{3,j}) = \begin{pmatrix} 0 & 1.25 & 6.25 \end{pmatrix}$$

Putting $(a_{3,j})$ to the left, we get $\underline{(a_{3,j}) = \frac{1}{4}(a_{1,j}) + (\bar{a}_{3,j})}$
and thus

$$\mathbf{A} = \begin{pmatrix} 4 & 3 & -1 \\ -2 & -4 & 5 \\ 1 & 2 & 6 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ -\frac{1}{2} & 1 & 0 \\ \frac{1}{4} & 0 & 1 \end{pmatrix} \begin{pmatrix} 4 & 3 & -1 \\ 0 & -2.5 & 4.5 \\ 0 & 1.25 & 6.25 \end{pmatrix}$$

LU Decomposition, example

$$\mathbf{A} = \begin{pmatrix} 4 & 3 & -1 \\ -2 & -4 & 5 \\ 1 & 2 & 6 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ -\frac{1}{2} & 1 & 0 \\ \frac{1}{4} & 0 & 1 \end{pmatrix} \begin{pmatrix} 4 & 3 & -1 \\ 0 & -2.5 & 4.5 \\ 0 & 1.25 & 6.25 \end{pmatrix}$$

Next, we "eliminate" $\bar{a}_{3,2} = 1.25$
by forming $(\bar{\bar{a}}_{3,j}) = (\bar{a}_{3,j}) + \frac{1}{2}(\bar{a}'_{2,j})$:

$$(\bar{\bar{a}}_{3,j}) = \begin{pmatrix} 0 & 0 & 8.5 \end{pmatrix}$$

$$(\bar{a}_{3,j}) = (a_{3,j}) - \tfrac{1}{4}(a_{1,j})$$

Expressing $(\bar{a}_{3,j})$ by rows that show up, we get
$(\bar{\bar{a}}_{3,j}) = (\bar{a}_{3,j}) + \frac{1}{2}(\bar{a}'_{2,j}) = (a_{3,j}) - \frac{1}{4}(a_{1,j}) + \frac{1}{2}(\bar{a}'_{2,j})$

Putting $(a_{3,j})$ to the left,
we get $(a_{3,j}) = \frac{1}{4}(a_{1,j}) - \frac{1}{2}(\bar{a}'_{2,j}) + (\bar{\bar{a}}_{3,j})$ and thus

$$\mathbf{A} = \begin{pmatrix} 4 & 3 & -1 \\ -2 & -4 & 5 \\ 1 & 2 & 6 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ -\frac{1}{2} & 1 & 0 \\ \frac{1}{4} & -\frac{1}{2} & 1 \end{pmatrix} \begin{pmatrix} 4 & 3 & -1 \\ 0 & -2.5 & 4.5 \\ 0 & 0 & 8.5 \end{pmatrix}$$

Construction with Operators (1)

Decomposition steps: construct L and U in parallel

– example:

$$\mathbf{A} = \begin{pmatrix} 1 & 0 & -2 & 3 \\ -1 & 2 & 1 & 1 \\ 0 & 0 & -3 & -2 \\ 1 & -2 & 2 & 0 \end{pmatrix}$$

$$\mathbf{A} = \mathbf{I}\,\mathbf{A} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & -2 & 3 \\ -1 & 2 & 1 & 1 \\ 0 & 0 & -3 & -2 \\ 1 & -2 & 2 & 0 \end{pmatrix}$$

– next: use $\mathbf{E}_1$ to eliminate all $a_{i>1,1}$ from **A** ...
(in parallel: adapt the left matrix, accordingly)

[see MATLAB example 2a_B]

Construction with Operators (2)

Decomposition steps: construct L and U in parallel

$$\mathbf{A} = \mathbf{I}\,\mathbf{A} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & -2 & 3 \\ -1 & 2 & 1 & 1 \\ 0 & 0 & -3 & -2 \\ 1 & -2 & 2 & 0 \end{pmatrix}$$

– next: use $\mathbf{E}_1$ to eliminate all $a_{i>1,1}$ from **A** …
(in parallel: adapt the left matrix, accordingly)

$$\mathbf{E}_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -\frac{-1}{1} & 1 & 0 & 0 \\ -\frac{0}{1} & 0 & 1 & 0 \\ -\frac{1}{1} & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{pmatrix} \qquad \mathbf{E}_1^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}$$

(more on inverse matrices later)

– with $\mathbf{E}_1$, we can now write $\mathbf{A} = \mathbf{I}\,\mathbf{A} = \mathbf{I}\,\underbrace{\mathbf{E}_1^{-1}\,\mathbf{E}_1}_{\mathbf{I}}\,\mathbf{A} = \underbrace{\mathbf{I}\,\mathbf{E}_1^{-1}}_{\mathbf{L}_1}\,\underbrace{\mathbf{E}_1\,\mathbf{A}}_{\mathbf{U}_1}$

- left-multiplication of **A** with $\mathbf{E}_1$
- right-multiplication of **I** with $\mathbf{E}_1^{-1}$ (only 1st column of **I** changes!)

Construction with Operators (3)

Decomposition steps: construct L and U in parallel

$$\mathbf{A} = \underbrace{\mathbf{I}\,\mathbf{E}_1^{-1}}_{\mathbf{L}_1}\,\underbrace{\mathbf{E}_1\,\mathbf{A}}_{\mathbf{U}_1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & -2 & 3 \\ 0 & 2 & -1 & 4 \\ 0 & 0 & -3 & -2 \\ 0 & -2 & 4 & -3 \end{pmatrix}$$

– next: use $\mathbf{E}_2$ to eliminate all $a_{i>2,2}$ from $\mathbf{U}_1$ …
(in parallel: adapt the left matrix, accordingly)

$$\mathbf{E}_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -\frac{0}{2} & 1 & 0 \\ 0 & -\frac{-2}{2} & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix} \quad \mathbf{E}_2^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{pmatrix}$$

– now we can write $\mathbf{A} = \mathbf{L}_1\,\mathbf{U}_1 = \mathbf{L}_1\,\underbrace{\mathbf{E}_2^{-1}\,\mathbf{E}_2}_{\mathbf{I}}\,\mathbf{U}_1 = \underbrace{\mathbf{L}_1\,\mathbf{E}_2^{-1}}_{\mathbf{L}_2}\,\underbrace{\mathbf{E}_2\,\mathbf{U}_1}_{\mathbf{U}_2}$

Construction with Operators (4)

Decomposition steps: construct L and U in parallel

$$\mathbf{A} = \underbrace{\mathbf{L}_1\,\mathbf{E}_2^{-1}}_{\mathbf{L}_2}\;\underbrace{\mathbf{E}_2\,\mathbf{U}_1}_{\mathbf{U}_2} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & -1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & -2 & 3 \\ 0 & 2 & -1 & 4 \\ 0 & 0 & -3 & -2 \\ 0 & 0 & 3 & 1 \end{pmatrix}$$

– next: use $\mathbf{E}_3$ to eliminate $a_{4,3}$ from $\mathbf{U}_2$ …
(in parallel: adapt the left matrix, accordingly)

$$\mathbf{E}_3 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{3}{-3} & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix} \quad \mathbf{E}_3^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{pmatrix}$$

– now we can write $\mathbf{A} = \mathbf{L}_2\,\mathbf{U}_2 = \mathbf{L}_2\,\underbrace{\mathbf{E}_3^{-1}\,\mathbf{E}_3}_{\mathbf{I}}\,\mathbf{U}_2 = \underbrace{\mathbf{L}_2\,\mathbf{E}_3^{-1}}_{\mathbf{L}}\,\underbrace{\mathbf{E}_3\,\mathbf{U}_2}_{\mathbf{U}}$

Construction with Operators (5)

Decomposition steps: construct L and U in parallel

$$\mathbf{A} = \underbrace{\mathbf{L}_2\,\mathbf{E}_3^{-1}}_{\mathbf{L}}\,\underbrace{\mathbf{E}_3\,\mathbf{U}_2}_{\mathbf{U}} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & -1 & -1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & -2 & 3 \\ 0 & 2 & -1 & 4 \\ 0 & 0 & -3 & -2 \\ 0 & 0 & 0 & -1 \end{pmatrix}$$

All steps combined:

$$\mathbf{A} = \mathbf{I}\,\mathbf{A} = \underbrace{\mathbf{I}\,\mathbf{E}_1^{-1}\ \ldots\ \mathbf{E}_{n-1}^{-1}}_{\mathbf{L}}\,\underbrace{\mathbf{E}_{n-1}\ \ldots\ \mathbf{E}_1\,\mathbf{A}}_{\mathbf{U}} = \underbrace{\mathbf{E}^{-1}}_{\mathbf{L}}\,\underbrace{\mathbf{E}\,\mathbf{A}}_{\mathbf{U}}$$

LU Decomposition with Permutation

What, if the targeted pivot-element is 0**?**

– then permutation helps:

– a permutation matrix $\mathbf{P}$
is an identity matrix $\mathbf{I}$ with swapped rows

– then $\mathbf{PA}$ is $\mathbf{A}$ with swapped rows,
example:

in MATLAB: „`[L, U, P] = lu(A)`" gives
the LU decomposition of $\mathbf{A}$ (with permutation)

– so given that a standard LU decomposition of $\mathbf{A}$ fails,
an appropriate $\mathbf{PA}$ can be decomposed (if $\mathbf{A}$ non-singular), thus

– $\mathbf{Ax} = \mathbf{b} \Rightarrow \mathbf{PAx} = \mathbf{Pb} \Rightarrow \mathbf{LUx} = \mathbf{Pb} \Rightarrow$
use $\mathbf{Pb}$ for substitution

Solve $\mathbf{A}\,\mathbf{x} = \mathbf{b}$ with the Inverse of A

Given the linear equation system $\mathbf{A}\,\mathbf{x} = \mathbf{b}$:

– solving $\mathbf{A}\,\mathbf{x} = \mathbf{b}$ for $\mathbf{x}$ means, in principle, to find $\mathbf{A}^{-1}$,
since $\mathbf{x} = \mathbf{A}^{-1}\,\mathbf{A}\,\mathbf{x} = \mathbf{A}^{-1}\,\mathbf{b}$

– usually,
finding $\mathbf{A}^{-1}$ is not easy… (if possible at all!)

– for some types of $\mathbf{A}$, however, finding $\mathbf{A}^{-1}$ is easy! :–)

Easy Inverses (easiest)

Unitary matrices M are easily inverted: $\mathbf{M}^{-1} = \mathbf{M}^{\mathsf{T}}$
(for complex numbers, $\mathbf{M}^{-1} = \mathbf{M}^*$)

- example 1: the identity matrix $\mathbf{I}$ $\Rightarrow$ $\mathbf{I}^{-1} = \mathbf{I}^{\mathsf{T}} = \mathbf{I}$
- example 2: all permutation matrices $\mathbf{P}$ $\Rightarrow$ $\mathbf{P}^{-1} = \mathbf{P}^{\mathsf{T}}$, f.i.:

$$\begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

- example 3: all orthonormal matrices $\mathbf{V}$ $\Rightarrow$ $\mathbf{V}^{-1} = \mathbf{V}^*$, since $\mathbf{v}_i^{\top}\mathbf{v}_j = \delta_{i,j}$

$$\underbrace{\begin{pmatrix} \mathbf{v}_1^{\top} \\ \mathbf{v}_2^{\top} \\ \vdots \\ \mathbf{v}_n^{\top} \end{pmatrix}}_{\mathbf{V}^{\mathsf{T}}} \underbrace{\begin{pmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \cdots & \mathbf{v}_n \end{pmatrix}}_{\mathbf{V}} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix}$$

Inverting (Undoing) An Elimination

As we can take out (a certain multitude of) **the pivot row, we also can add** (a certain multitude of) **it back in:**

- For example:

$$\mathbf{A}' = \mathbf{E}_{3,2}\,\mathbf{A} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -2 & 1 \end{pmatrix} \begin{pmatrix} 1 & 2 & 3 \\ 0 & 1 & 4 \\ 0 & 2 & 5 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 3 \\ 0 & 1 & 4 \\ 0 & 0 & -3 \end{pmatrix}$$

  Component $a_{3,2}$ was eliminated by subtracting the pivot-row ($\mathbf{A}$'s 2^{nd} row) twice from row #3.

  To get back to $\mathbf{A}$, do the exactly the opposite, i.e., we add the pivot-row back in (also twice):

$$\mathbf{A} = \mathbf{E}_{3,2}^{-1}\,\mathbf{A}' = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 2 & 1 \end{pmatrix} \begin{pmatrix} 1 & 2 & 3 \\ 0 & 1 & 4 \\ 0 & 0 & -3 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 3 \\ 0 & 1 & 4 \\ 0 & 2 & 5 \end{pmatrix}$$

Easy Inverses (very easy)

Straight-forward inversion of elimination matrices:

$$\mathbf{E}_{i,j} = \begin{pmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \cdots & 1 & \cdots & 0 & \cdots & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & \cdots & -\frac{a_{i,j}}{a_{j,j}} & \cdots & 1 & \cdots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{pmatrix} \qquad \mathbf{E}_{i,j}^{-1} = \begin{pmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \cdots & 1 & \cdots & 0 & \cdots & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & \cdots & +\frac{a_{i,j}}{a_{j,j}} & \cdots & 1 & \cdots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{pmatrix}$$

$$\mathbf{E}_{j} = \begin{pmatrix} 1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & & \vdots \\ 0 & \cdots & 1 & 0 & \cdots & 0 \\ 0 & \cdots & -\frac{a_{j+1,j}}{a_{j,j}} & 1 & \cdots & 0 \\ \vdots & & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & -\frac{a_{n,j}}{a_{j,j}} & 0 & \cdots & 1 \end{pmatrix} \qquad \mathbf{E}_{j}^{-1} = \begin{pmatrix} 1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & & \vdots \\ 0 & \cdots & 1 & 0 & \cdots & 0 \\ 0 & \cdots & +\frac{a_{j+1,j}}{a_{j,j}} & 1 & \cdots & 0 \\ \vdots & & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & +\frac{a_{n,j}}{a_{j,j}} & 0 & \cdots & 1 \end{pmatrix}$$

– i.e., $\mathbf{E}_{i,j}{}^{-1} = 2\,\mathbf{I} - \mathbf{E}_{i,j}$ and $\mathbf{E}_{j}{}^{-1} = 2\,\mathbf{I} - \mathbf{E}_{j}$

Easy Inverses (easy)

Triangular matrices (upper & lower) **are easy to invert**:

– we can use Gauss-Jordan elimination to invert triangular matrices:

$$\mathbf{U} = \begin{pmatrix} 1 & 2 & 3 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix} \Rightarrow$$

use appropriate elimination operations $\hat{\mathbf{E}}_k$ to transform **U** into **I**

– iterative elimination transforms **U** into **I**:

$$\hat{\mathbf{E}}_2 \cdots \hat{\mathbf{E}}_n \mathbf{U} = \mathbf{I}$$

– all elimination matrices multiplied are then, of course, the inverse of **U**:

$$\mathbf{U}^{-1} = \hat{\mathbf{E}} = \hat{\mathbf{E}}_2 \cdots \hat{\mathbf{E}}_n$$

$$\mathbf{U}^{-1} = \begin{pmatrix} 1 & -2 & -1 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\left(\begin{array}{ccc|ccc} 1 & 2 & 3 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{array}\right)$$

$$\left(\begin{array}{ccc|ccc} 1 & 2 & 0 & 1 & 0 & -3 \\ 0 & 1 & 0 & 0 & 1 & -1 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{array}\right)$$

$$\left(\begin{array}{ccc|ccc} 1 & 0 & 0 & 1 & -2 & -1 \\ 0 & 1 & 0 & 0 & 1 & -1 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{array}\right)$$

Inverting General Matrices

**Many methods exist to invert matrices;
one (based on the above) is:**

- first decompose $\mathbf{A}$ into $\mathbf{L}\,\mathbf{U}$ (LU decomposition)
- then invert $\mathbf{L}$ and $\mathbf{U}$ (easy: Gauss-Jordan elimination)
- then recompose them to get the inverse of $\mathbf{A}$:

$$\mathbf{A}^{-1} = (\mathbf{L}\,\mathbf{U})^{-1} = \mathbf{U}^{-1}\,\mathbf{L}^{-1}$$

Singular A (System not Solvable?)

What if A (in $\mathbf{A}\,\mathbf{x} = \mathbf{b}$) **is singular** (not invertible)?

- depending on **b**, there
  - is either no solution at all
  - or infinitely many of them!

$$\mathbf{A} = \begin{pmatrix} 1 & 2 & 3 \\ 0 & 1 & -1 \\ 1 & 0 & 5 \end{pmatrix} \quad \text{with} \quad \det \mathbf{A} = 0$$

Let's try and solve $\mathbf{A}\mathbf{x} = \mathbf{b}_1 = (1\ 2\ 3)^\top$ for $\mathbf{x}$:

$$\left(\begin{array}{ccc|c} 1 & 2 & 3 & 1 \\ 0 & 1 & -1 & 2 \\ 1 & 0 & 5 & 3 \end{array}\right)$$

Rewriting row 3, $(\bar{a}_{3,j}) = (a_{3,j}) - (a_{1,j})$:

$$\left(\begin{array}{ccc|c} 1 & 2 & 3 & 1 \\ 0 & 1 & -1 & 2 \\ 0 & -2 & 2 & 2 \end{array}\right)$$

Trying another step, $(\bar{\bar{a}}_{3,j}) = (\bar{a}_{3,j}) + 2(a_{2,j})$:

$$\left(\begin{array}{ccc|c} 1 & 2 & 3 & 1 \\ 0 & 1 & -1 & 2 \\ 0 & 0 & 0 & 6 \end{array}\right)$$

The last equation $(0 = 6)$ cannot be solved, i.e., $\mathbf{A}\mathbf{x} = \mathbf{b}_1$ does not have any solution.

Singular A

What if A (in **A x = b**) **is singular (not invertible)?**

- depending on **b**, there
  - is either no solution at all
  - or infinitely many of them!

$$\mathbf{A} = \begin{pmatrix} 1 & 2 & 3 \\ 0 & 1 & -1 \\ 1 & 0 & 5 \end{pmatrix} \quad \text{with} \quad \det \mathbf{A} = 0$$

Now, let's try to solve $\mathbf{Ax} = \mathbf{b}_2 = (1 \; 2 \; -3)^\top$ for $\mathbf{x}$:

$$\left(\begin{array}{ccc|c} 1 & 2 & 3 & 1 \\ 0 & 1 & -1 & 2 \\ 1 & 0 & 5 & -3 \end{array}\right)$$

Rewriting row 3, $(\bar{a}_{3,j}) = (a_{3,j}) - (a_{1,j})$:

$$\left(\begin{array}{ccc|c} 1 & 2 & 3 & 1 \\ 0 & 1 & -1 & 2 \\ 0 & -2 & 2 & -4 \end{array}\right)$$

Trying another step, $(\bar{\bar{a}}_{3,j}) = (\bar{a}_{3,j}) + 2(a_{2,j})$:

$$\left(\begin{array}{ccc|c} 1 & 2 & 3 & 1 \\ 0 & 1 & -1 & 2 \\ 0 & 0 & 0 & 0 \end{array}\right)$$

Now the last equation $(0 = 0)$ is a tautology, i.e., $\mathbf{Ax} = \mathbf{b}_2$ has infinitely many solutions.

Singular A

What if **A** (in **A x = b**) is singular (not invertible)?

– depending on **b**, there
  - is either no solution at all
  - or infinitely many of them!

$$\mathbf{A} = \begin{pmatrix} 1 & 2 & 3 \\ 0 & 1 & -1 \\ 1 & 0 & 5 \end{pmatrix} \quad \text{with} \quad \det \mathbf{A} = 0$$

Now, let's try to solve $\mathbf{A}\mathbf{x} = \mathbf{b}_2 = (1\ 2\ -3)^\top$ for $\mathbf{x}$:

$$\left(\begin{array}{ccc|c} 1 & 2 & 3 & 1 \\ 0 & 1 & -1 & 2 \\ 1 & 0 & 5 & -3 \end{array}\right)$$

Rewriting row 3, $(\bar{a}_{3,j}) = (a_{3,j}) - (a_{1,j})$:

$$\left(\begin{array}{ccc|c} 1 & 2 & 3 & 1 \\ 0 & 1 & -1 & 2 \\ 0 & -2 & 2 & -4 \end{array}\right)$$

Trying another step, $(\bar{\bar{a}}_{3,j}) = (\bar{a}_{3,j}) + 2(a_{2,j})$:

$$\left(\begin{array}{ccc|c} 1 & 2 & 3 & 1 \\ 0 & 1 & -1 & 2 \\ 0 & 0 & 0 & 0 \end{array}\right)$$

Now the last equation $(0 = 0)$ is a tautology, i.e., $\mathbf{A}\mathbf{x} = \mathbf{b}_2$ has infinitely many solutions.

Which?

The last row did not resolve z (so we set $z = t$).
The middle row reads $y - z = 2$ or $y = 2 + t$.
Then, the top row is $x + 2y + 3z = x + 4 + 2t + 3t = 1$, i.e., $x = -5t - 3$

Accordingly, all points on the one-parameter line

$$\mathbf{x} = \begin{pmatrix} -5t - 3 \\ t + 2 \\ t \end{pmatrix}$$

solve $\mathbf{A}\mathbf{x} = \mathbf{b}_2$.

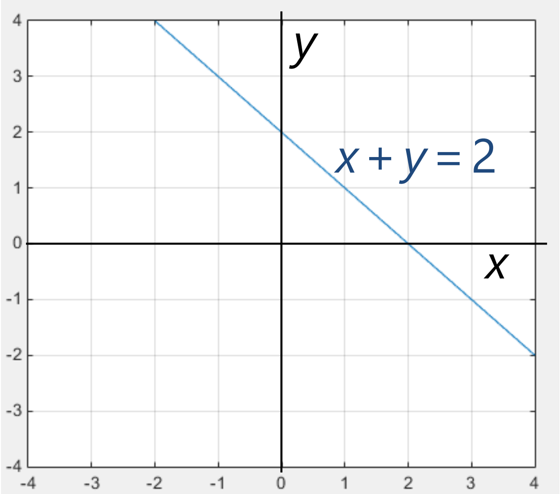
Test:

$$\begin{array}{llllll} \text{equ. 1:} & x & +2y & +3z & = -5t - 3 + 4 + 2t + 3t = & 1 \\ \text{equ. 2:} & & y & -z & = 2 + t - t = & 2 \\ \text{equ. 3:} & x & & +5z & = -5t - 3 + 5t = & -3 \end{array}$$

Illustration

Any linear equation involving $x_1, x_2, \ldots, x_n$

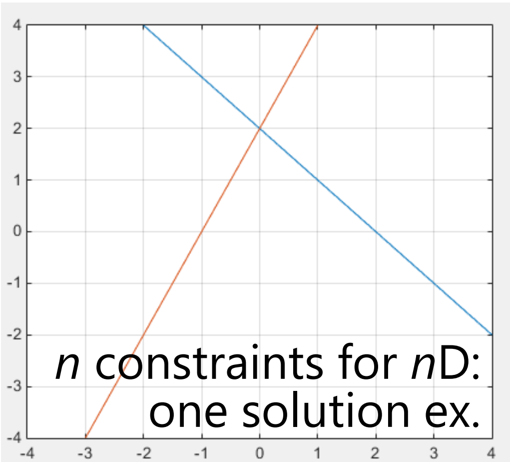
- can be seen as a linear constraint on $x_1, x_2, \ldots, x_n$
- confining $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T$ to an $(n\text{-}1)$-dimensional, linear subspace
- example:
  - $x + y = 2$ confines $\mathbf{x} = (x, y)^T \in R^2$ to a line
  - $x + y + z = 2$ confines $\mathbf{x} = (x, y, z)^T \in R^3$ to a plane (in 3D)

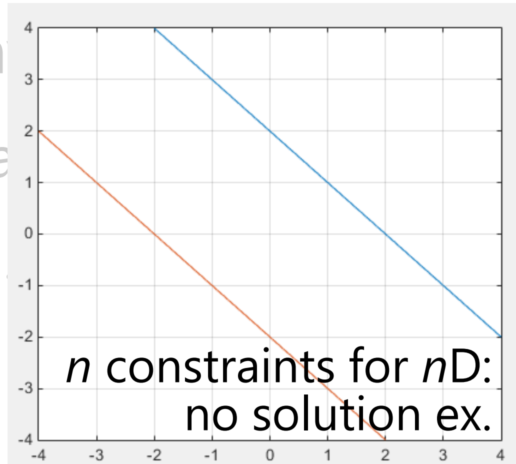


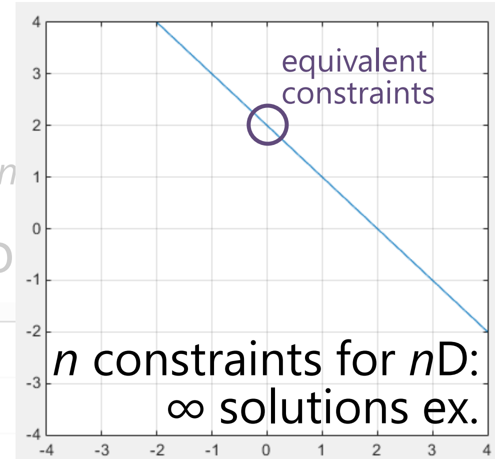
Multiple linear equations

- invoke multiple constraints—can result in
  - a system with one solution,
    no solution,
    or infinitely many solutions

Illustration







- $x + y + z = 2$ confines $\mathbf{x} = (x, y, z)^T \in R^3$ to a plane (in 3D)

Multiple linear equations

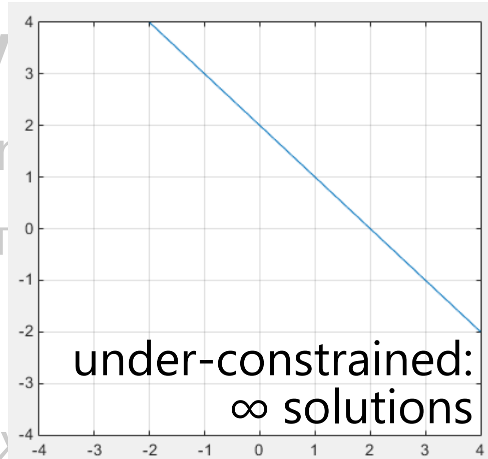
– invoke multiple constraints—can result in

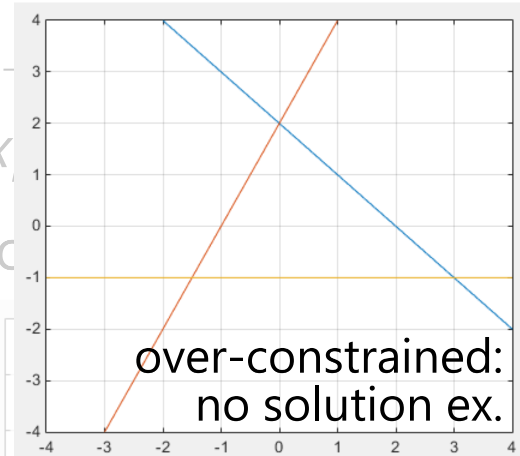
- a system with one solution,
  no solution,
  or infinitely many solutions

Illustration

Any linear equation involv

- can be seen as a linear co
- confining $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T$
- example:
  - $x + y = 2$ confines $\mathbf{x} = ($
  - $x + y + z = 2$ confines $\mathbf{x} = (x, y, z)^T \in R^3$ to a plane (in 3D)





Multiple linear equations

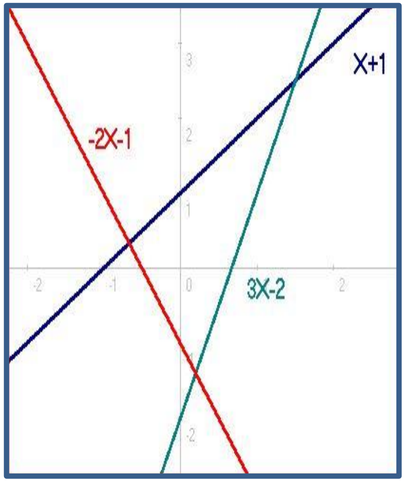
- invoke multiple constraints—can result
  - a system with one solution, no solution, or ∞ solutions
  - an under-constrained system: n constraints in mD, $n<m$
  - an over-constrained system: n constraints in mD, $n>m$

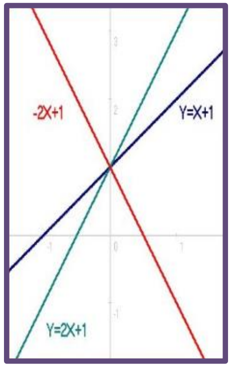
Non-square Matrices

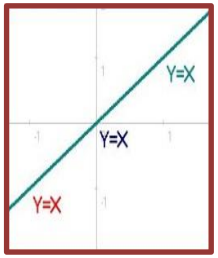
In $\mathbf{A\,x = b}$, $\mathbf{A}$ is an n x m **matrix** (and $\mathbf{x}$ is mD and $\mathbf{b}$ is nD):

– if $n > m$, then $\mathbf{A\,x = b}$ is an over-determined system

- usually no solution to $\mathbf{x}$
- in exceptional cases:
  one solution,
  or even ∞

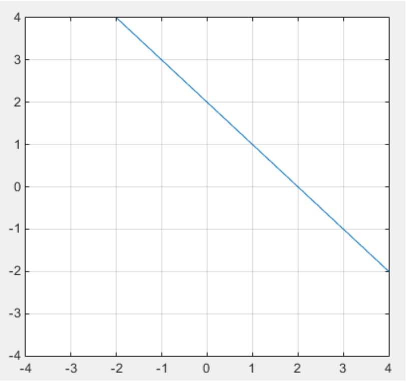


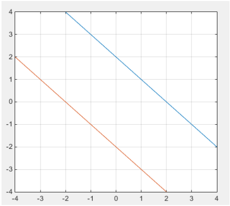




– if $n < m$, then $\mathbf{A\,x = b}$ is an under-determined system

- usually infinitely many solutions
- in exceptional cases:
  no solution





Non-square Matrices

In $\mathbf{A}\mathbf{x}=\mathbf{b}$, A is an *n* x *m* **matrix** (and **x** is *m*D and **b** is *n*D):

– if $n \neq m$, i.e., **A** is non-square,

- **A** cannot be inverted, of course
- **A** can be "somehow" LU decomposed (LUP decomp.), however!

Case $m>n$:

$$\mathbf{A} = \begin{pmatrix} 1 & 2 & 3 \\ 1 & 0 & -2 \end{pmatrix} \quad \Rightarrow \quad \mathbf{A} = \mathbf{LU} = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 2 & 3 \\ 0 & -2 & -5 \end{pmatrix}$$

$$\mathbf{Ax} = \begin{pmatrix} 1 & 2 & 3 \\ 0 & -2 & -5 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} = \mathbf{b}$$

$$\mathbf{Ax} = \mathbf{LUx} = \mathbf{b} \quad \rightarrow \quad \mathbf{Ux} = \mathbf{y} \quad \text{and} \quad \mathbf{Ly} = \mathbf{b}$$

$\mathbf{Ly} = \mathbf{b}$ can be (usually) solved for $\mathbf{y}$, leaving an under-determined, triangular system $\mathbf{Ux} = \mathbf{y}$ that can be "solved" on the basis of parameters.

Non-square Matrices

In $\mathbf{A}\mathbf{x} = \mathbf{b}$, $\mathbf{A}$ is an *n* x *m* **matrix** (and **x** is *m*D and **b** is *n*D):

- if $n \neq m$, i.e., **A** is non-square,
  - **A** cannot be inverted, of course
  - **A** can be "somehow" LU decomposed (LUP decomp.), however!

Case $n>m$:

$$\mathbf{A} = \begin{pmatrix} 1 & 1 \\ 2 & 0 \\ 3 & -2 \end{pmatrix} \quad \Rightarrow \quad \mathbf{A} = \mathbf{L}\mathbf{U} = \begin{pmatrix} 1 & 0 \\ 1/3 & 1 \\ 2/3 & 4/5 \end{pmatrix} \begin{pmatrix} 3 & -2 \\ 0 & 5/3 \end{pmatrix}$$

$$\mathbf{A}\mathbf{x} = \begin{pmatrix} 1 & 1 \\ 2 & 0 \\ 3 & -2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} = \mathbf{b}$$

$$\mathbf{A}\mathbf{x} = \mathbf{L}\mathbf{U}\mathbf{x} = \mathbf{b} \quad \rightarrow \quad \mathbf{U}\mathbf{x} = \mathbf{y} \quad \text{and} \quad \mathbf{L}\mathbf{y} = \mathbf{b}$$

Usually, $\mathbf{L}\mathbf{y} = \mathbf{b}$ cannot be solved.

Outlook

Next time: vector spaces

- basis of a vector space
- standard basis
- basis of a sub-space
- coordinates wrt. a basis
- changing the basis
- changing the basis of a mapping
- projections

Thereafter:

- singular value decomposition (SVD)
- eigenanalysis
- principal component analysis (PCA)

Reading and Related Material

In the book:

- chapter 2 (on linear systems), mostly section 2.1, and related parts
- chapter 15 (on SVD, etc.): coming soon! :–)

Course notes:

- section 4

An interactive online-book:

- http://ImmersiveMath.com/,
  chapters (1–4 &) 5 (on Gaussian elimination)
  and 6 (on The Matrix)

On Wikipedia:

- many good pages