What is a tensor? (Part I)

Lek-Heng Lim

University of Chicago

goals of these tutorials

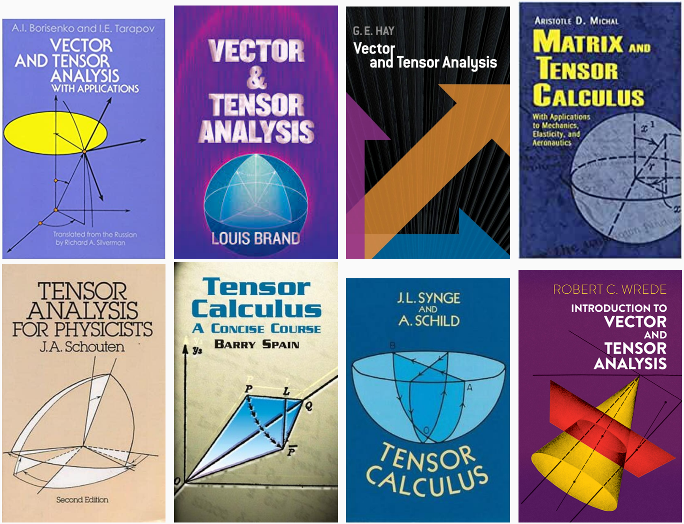
- very elementary introduction to tensors
- through the lens of linear algebra and numerical linear algebra
- highlight their roles in computations
- trace chronological development through three definitions
  1. a multi-indexed object that satisfies tensor transformation rules
  2. a multilinear map
  3. an element of a tensor product of vector spaces
- all three definitions remain useful today

tensors via transformation rules

earliest definition

- trickiest among the three definitions
- first appearance in Woldemar Voigt's 1898 book on crystallography
  - "An abstract entity represented by an array of components that are functions of coordinates such that, under a transformation of cooordinates, the new components are related to the transformation and to the original components in a definite way"
- main issue: defines an entity by giving its change-of-bases formulas but without specifying the entity itself
- likely reason for notoriety of tensors as a difficult subject to master
  - J. Earman, C. Glymour, "Lost in tensors: Einstein's struggles with covariance principles 1912–1916," *Stud. Hist. Phil. Sci.*, **9** (1978), no. 4, pp. 251–278

definition in Dover books c. 1950s



- “a multi-indexed object that satisfies certain transformation rules”

fortunately for us

- linear algebra as we know it today was a subject in its infancy when Einstein was trying to learn tensors
- vector space, linear map, dual space, basis, change-of-basis, matrix, matrix multiplication, etc, were all obscure notions back then
  - 1858: 3×3 matrix product[1] (Cayley)
  - 1888: vector space and $n \times n$ matrix product (Peano)
  - 1898: tensor (Voigt)
- we enjoy the benefit of a hundred years of pedagogical progress
- next slides: look at tensor transformation rules in light of linear algebra and numerical linear algebra

[1] we still do not know the optimal algorithm for this

eigen and singular values

- **eigenvalue and vectors:** $A \in \mathbb{C}^{n\times n}$, $Av = \lambda v$, for any invertible $X \in \mathbb{C}^{n\times n}$,

$$(XAX^{-1})Xv = \lambda Xv$$

  - eigenvalue $\lambda' = \lambda$, eigenvector $v' = Xv$, and $A' = XAX^{-1}$

- **singular values and vectors:** $A \in \mathbb{R}^{m\times n}$,

$$\begin{cases} Av = \sigma u, \\ A^\top u = \sigma v \end{cases}$$

  for any orthogonal $X \in \mathbb{R}^{m\times m}$, $Y \in \mathbb{R}^{n\times n}$,

$$\begin{cases} (XAY^\top)Yv = \sigma Xu, \\ (XAY^\top)^\top Xu = \sigma Yv \end{cases}$$

  - singular value $\sigma' = \sigma$, left singular vector $u' = Xu$, left singular vector $v' = Yv$, and $A' = XAY^\top$

matrix product and linear systems

- **matrix product:** $A \in \mathbb{C}^{m\times n}$, $B \in \mathbb{C}^{n\times p}$, $C \in \mathbb{C}^{m\times p}$, $AB = C$, for any invertible X, Y, Z,

$$(XAY^{-1})(YBZ^{-1}) = XCZ^{-1}$$

  - $A' = XAY^{-1}$, $B' = YBZ^{-1}$, $C' = XCZ^{-1}$
- **linear system:** $A \in \mathbb{C}^{m\times n}$, $b \in \mathbb{C}^{m}$, $Av = b$, for any invertible X, Y,

$$(XAY^{-1})(Yv) = Xb$$

  - $A' = XAY^{-1}$, $b' = Xb$, $v' = Yv$

ordinary and total least squares

- **ordinary least squares:** $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$,

$$\min_{v \in \mathbb{R}^n} \|Av - b\|^2 = \min_{v \in \mathbb{R}^n} \|(XAY^{-1})Yv - Xb\|^2$$

  for any orthogonal $X \in \mathbb{R}^{m \times m}$ and invertible $Y \in \mathbb{R}^{n \times n}$
  - $A' = XAY^{-1}$, $b' = Xb$, $v' = Yv$, minimum value $\rho' = \rho$
- **total least squares:** $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$, then

$$\begin{aligned} &\min \{\|E\|^2 + \|r\|^2 : (A+E)v = b + r\} \\ &\quad = \min \{\|XEY^\top\|^2 + \|Xr\|^2 : (XAY^\top + XEY^\top)Yv = Xb + Xr\} \end{aligned}$$

  for any orthogonal $X \in \mathbb{R}^{m \times m}$ and orthogonal $Y \in \mathbb{R}^{n \times n}$
  - $A' = XAY^\top$, $E' = XEY^\top$, $b' = Xb$, $r' = Xr$, $v' = Yv$

rank, norm, determinant, intertia

- **rank, norm, determinant:** $A \in \mathbb{R}^{m \times n}$

  $$\text{rank}(XAY^{-1}) = \text{rank}(A), \quad \det(XAY^{-1}) = \det(A), \quad \|XAY^{-1}\| = \|A\|$$

  for X and Y invertible, special linear, or orthogonal, respectively
  - determinant identically zero whenever $m \neq n$
  - $\| \cdot \|$ either spectral, nuclear, or Frobenius norm
- **positive definiteness:** $A \in \mathbb{R}^{n \times n}$ positive definite iff

  $$XAX^{\mathsf{T}} \qquad \text{or} \qquad X^{-\mathsf{T}}AX^{-1}$$

  positive definite for any invertible $X \in \mathbb{R}^{n \times n}$

observation

- almost everything we study in linear algebra and numerical linear algebra satisfies tensor transformation rules
- different names, same thing:
  - equivalence of matrices: $A' = XAY^{-1}$
  - similarity of matrices: $A' = XAX^{-1}$
  - congruence of matrices: $A' = XAX^{\mathsf{T}}$
- almost everything we study in linear algebra and numerical linear algebra is about 0-, 1-, 2-tensors

triply ambiguous

- transformation rules may mean different things

$$A' = XAY^{-1}, \quad A' = XAY^{\mathsf{T}}, \quad A' = XAX^{-1}, \quad A' = XAX^{\mathsf{T}}$$

  and more

- matrices in transformation rules may have different properties

$$X \in \mathrm{GL}(n),\ \mathrm{SL}(n),\ \mathrm{O}(n),$$
$$(X, Y) \in \mathrm{GL}(m) \times \mathrm{GL}(n),\ \mathrm{SL}(m) \times \mathrm{SL}(n),\ \mathrm{O}(m) \times \mathrm{O}(n),\ \mathrm{O}(m) \times \mathrm{GL}(n)$$

  and more

- alternative (but equivalent) forms just as common

$$A' = X^{-1}AY, \quad A' = X^{-1}AY^{-\mathsf{T}}, \quad A' = X^{-1}AX, \quad A' = X^{-1}AX^{-\mathsf{T}}$$

math perspective

- multi-indexed object $\lambda \in \mathbb{R}$, $v \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$, etc, represents the tensor
- transformation rule $A' = XAY^{-1}$, $A' = XAY^{-1}$, $A' = XAX^{\mathsf{T}}$, etc, defines the tensor
- but the tensor has been left unspecified
- easily fixed with modern definitions ② and ③
- need a context in order to use definition ①
- is $A = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \\ 3 & 4 & 5 \end{bmatrix}$ a tensor?
- it is a tensor if we are interested in, say, its eigenvalues and eigenvectors, in which case A transforms as a mixed 2-tensor

physics perspective

- remember definition ① came from physics — they don't ask
  - what is a tensor?

  but
  - is stress a tensor?
  - is deformation a tensor?
  - is electromagnetic field strength a tensor?
- unspecified quantity is placeholder for physical quantity like stress, deformation, etc
- it is a tensor if the multi-indexed object satisfies transformation rules under change-of-coordinates, i.e., definition ①
- makes perfect sense in a physics context
- is $A = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \\ 3 & 4 & 5 \end{bmatrix}$ a tensor?
- it is a tensor if it represents, say, stress, in which case A transforms as a contravariant 2-tensor

0-, 1-, 2-tensor transformation rules

contravariant 1-tensor:	$a' = X^{-1}a$	$a' = Xa$
covariant 1-tensor:	$a' = X^{\mathsf{T}}a$	$a' = X^{-\mathsf{T}}a$
covariant 2-tensor:	$A' = X^{\mathsf{T}}AX$	$A' = X^{-\mathsf{T}}AX^{-1}$
contravariant 2-tensor:	$A' = X^{-1}AX^{-\mathsf{T}}$	$A' = XAX^{\mathsf{T}}$
mixed 2-tensor:	$A' = X^{-1}AX$	$A' = XAX^{-1}$
contravariant 2-tensor:	$A' = X^{-1}AY^{-\mathsf{T}}$	$A' = XAY^{\mathsf{T}}$
covariant 2-tensor:	$A' = X^{\mathsf{T}}AY$	$A' = X^{-\mathsf{T}}AY^{-1}$
mixed 2-tensor:	$A' = X^{-1}AY$	$A' = XAY^{-1}$

multilinear matrix multiplication

- $A \in \mathbb{R}^{n_1 \times \cdots \times n_d}$
- $X \in \mathbb{R}^{m_1 \times n_1}, Y \in \mathbb{R}^{m_2 \times n_2}, \ldots, Z \in \mathbb{R}^{m_d \times n_d}$
- define
$$(X, Y, \ldots, Z) \cdot A = B$$
where $B \in \mathbb{R}^{m_1 \times \cdots \times m_d}$ given by
$$b_{i_1 \cdots i_d} = \sum_{j_1=1}^{n_1} \sum_{j_2=1}^{n_2} \cdots \sum_{j_d=1}^{n_d} x_{i_1 j_1} y_{i_2 j_2} \cdots z_{i_d j_d} a_{j_1 \cdots j_d}$$
- $d = 1$: reduces to $Xa = b$ for $a \in \mathbb{R}^n$, $b \in \mathbb{R}^m$
- $d = 2$: reduces to
$$(X, Y) \cdot A = XAY^{\mathsf{T}}$$

higher-order transformation rules 1

- $X_1 \in \mathrm{GL}(n_1), X_2 \in \mathrm{GL}(n_2), \dots, X_d \in \mathrm{GL}(n_d)$
- covariant d-tensor transformation rule:

$$A' = (X_1^{\mathsf{T}}, X_2^{\mathsf{T}}, \dots, X_d^{\mathsf{T}}) \cdot A$$

- contravariant d-tensor transformation rule:

$$A' = (X_1^{-1}, X_2^{-1}, \dots, X_d^{-1}) \cdot A$$

- mixed d-tensor transformation rule:

$$A' = (X_1^{-1}, \dots, X_p^{-1}, X_{p+1}^{\mathsf{T}}, \dots, X_d^{\mathsf{T}}) \cdot A$$

- contravariant order p, covariant order $d - p$, or type $(p, d - p)$

higher-order transformation rules 2

- when $n_1 = n_2 = \cdots = n_d = n$, $X \in \mathrm{GL}(n)$
- covariant d-tensor transformation rule:

$$A' = (X^{\mathsf{T}}, X^{\mathsf{T}}, \dots, X^{\mathsf{T}}) \cdot A$$

- contravariant d-tensor transformation rule:

$$A' = (X^{-1}, X^{-1}, \dots, X^{-1}) \cdot A$$

- mixed d-tensor transformation rule:

$$A' = (X^{-1}, \dots, X^{-1}, X^{\mathsf{T}}, \dots, X^{\mathsf{T}}) \cdot A$$

- getting ahead of ourselves, with definition ②, difference is between multilinear

$$f : \mathbb{V}_1 \times \cdots \times \mathbb{V}_d \to \mathbb{R} \quad \text{and} \quad f : \mathbb{V} \times \cdots \times \mathbb{V} \to \mathbb{R}$$

change-of-cooordinates matrices

- $X_1, \dots, X_d$ or X may belong to:

$$\begin{aligned}
\mathrm{GL}(n) &= \{X \in \mathbb{R}^{n\times n} : \det(X) \neq 0\} \\
\mathrm{SL}(n) &= \{X \in \mathbb{R}^{n\times n} : \det(X) = 1\} \\
\mathrm{O}(n) &= \{X \in \mathbb{R}^{n\times n} : X^{\mathsf{T}} X = I\}, \\
\mathrm{SO}(n) &= \{X \in \mathbb{R}^{n\times n} : X^{\mathsf{T}} X = I,\ \det(X) = 1\} \\
\mathrm{U}(n) &= \{X \in \mathbb{C}^{n\times n} : X^* X = I\} \\
\mathrm{SU}(n) &= \{X \in \mathbb{C}^{n\times n} : X^* X = I,\ \det(X) = 1\} \\
\mathrm{O}(p,q) &= \{X \in \mathbb{R}^{n\times n} : X^{\mathsf{T}} I_{p,q} X = I_{p,q}\} \\
\mathrm{SO}(p,q) &= \{X \in \mathbb{R}^{n\times n} : X^{\mathsf{T}} I_{p,q} X = I_{p,q},\ \det(X) = 1\} \\
\mathrm{Sp}(2n,\mathbb{R}) &= \{X \in \mathbb{R}^{2n\times 2n} : X^{\mathsf{T}} J X = J\} \\
\mathrm{Sp}(2n) &= \{X \in \mathbb{C}^{2n\times 2n} : X^{\mathsf{T}} J X = J,\ X^* X = I\}
\end{aligned}$$

- $I := I_n$ is $n \times n$ identity, $I_{p,q} := \begin{bmatrix} I_p & 0 \\ 0 & -I_q \end{bmatrix} \in \mathbb{R}^{n\times n}$, $J := \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix} \in \mathbb{R}^{2n\times 2n}$

change-of-cooordinates matrices

- again getting ahead of ourselves with definitions ② or ③,
  - if vector spaces involve have no extra structure, then $\mathrm{GL}(n)$
  - if inner product spaces, then $\mathrm{O}(n)$
  - if equipped with yet other structures, then whatever group that preserves those structures
- e.g., $\mathbb{R}^4$ equipped with Euclidean inner product:

$$\langle x, y\rangle = x_0 y_0 + x_1 y_1 + x_2 y_2 + x_3 y_3$$

  want $X \in \mathrm{O}(4)$ or $\mathrm{SO}(4)$
- e.g., $\mathbb{R}^4$ equipped with Lorentzian scalar product,

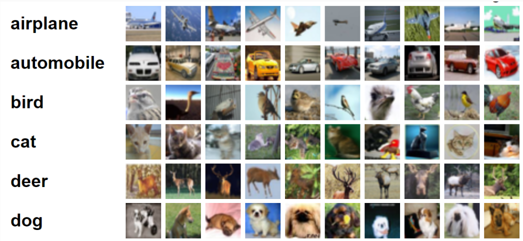
$$\langle x, y\rangle = x_0 y_0 - x_1 y_1 - x_2 y_2 - x_3 y_3,$$

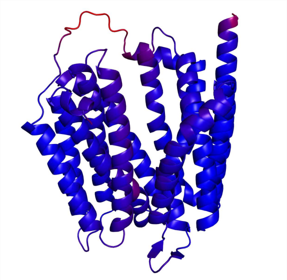
  want $X \in \mathrm{O}(1, 3)$ or $\mathrm{SO}(1, 3)$
- called **Cartesian tensors** or **Lorentzian tensors** respectively

transformation rule is the crux

why important (in machine learning)

- tensor transformation rules in modern parlance: equivariance
- exact same idea in equivariant neural network used by Google's **AlphaFold 2** [Jumper et al, 2020]





why important (in physics)

- special relativity is essentially the observation that the laws of physics are invariant under Lorentz transformations in $O(1,3)$ [Einstein, 1920]
- transformation rules under $O(1,3)$-analogue of Givens rotations:

$$\begin{bmatrix} \cosh\theta & -\sinh\theta & 0 & 0 \\ -\sinh\theta & \cosh\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} \cosh\theta & 0 & -\sinh\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sinh\theta & 0 & \cosh\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} \cosh\theta & 0 & 0 & -\sinh\theta \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -\sinh\theta & 0 & 0 & \cosh\theta \end{bmatrix}$$

  enough to derive most standard results of special relativity
- "Geometric Principle: The laws of physics must all be expressible as geometric (coordinate independent and reference frame independent) relationships between geometric objects (scalars, vectors, tensors, ...) that represent physical entitities." [Thorne, 1973]

why important (in mathematics)

- deriving higher-order tensorial analogues not a matter of just adding more indices to

$$\sum_{j=1}^{n} a_{ij} x_j = b_i, \quad \sum_{j=1}^{n} a_{ij} x_j = \lambda x_i, \quad \sum_{\sigma \in \mathfrak{S}_n} \operatorname{sgn}(\sigma) \prod_{i=1}^{n} a_{i\sigma(i)}$$

- need to satisfy tensor transformation rules
- e.g., $A \in \mathbb{R}^{2\times 2\times 2}$ has hyperdeterminant

$$\begin{aligned}
\det(A) = {} & a_{000}^2 a_{111}^2 + a_{001}^2 a_{110}^2 + a_{010}^2 a_{101}^2 + a_{011}^2 a_{100}^2 \\
& - 2(a_{000}a_{001}a_{110}a_{111} + a_{000}a_{010}a_{101}a_{111} + a_{000}a_{011}a_{100}a_{111} \\
& + a_{001}a_{010}a_{101}a_{110} + a_{001}a_{011}a_{110}a_{100} + a_{010}a_{011}a_{101}a_{100}) \\
& + 4(a_{000}a_{011}a_{101}a_{110} + a_{001}a_{010}a_{100}a_{111}),
\end{aligned}$$

- preserved by transformation $A' = (X, Y, Z) \cdot A$ for $X, Y, Z \in \mathrm{SL}(2)$
- just as determinant preserved by $A' = XAY^{\mathsf{T}}$ for $X, Y \in \mathrm{SL}(n)$

tensor multiplication?

- Hadamard product:

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \circ \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} a_{11}b_{11} & a_{12}b_{12} \\ a_{21}b_{21} & a_{22}b_{22} \end{bmatrix}$$

- seems a lot more obvious than standard matrix product

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} a_{11}b_{11} + a_{12}b_{21} & a_{11}b_{12} + a_{12}b_{22} \\ a_{21}b_{11} + a_{22}b_{21} & a_{21}b_{12} + a_{22}b_{22} \end{bmatrix}$$

- matrix product satisfies transformation rule for mixed 2-tensors $(XAY^{-1})(YBZ^{-1}) = X(AB)Z^{-1}$, i.e., defined on tensors
- Hadamard product undefined on tensors — depends on coordinates
- product on $\mathbb{R}^{m \times n \times p}$ or $\mathbb{R}^{n \times n \times n}$ that satisfies 3-tensor transformation rules does not exist

identity tensor?

- identity matrix $I \in \mathbb{R}^{3\times 3}$

$$I = \sum_{i=1}^{3} e_i \otimes e_i \in \mathbb{R}^{3\times 3}$$

  with $e_1, e_2, e_3 \in \mathbb{R}^3$ standard basis vectors
- $(Q, Q) \cdot I = QIQ^\intercal = I$ for any $Q \in \mathrm{O}(3)$, unique up to scalar multiples
- I is a Cartesian 2-tensor
- analogue in $\mathbb{R}^{3\times 3\times 3}$ is not

$$A = \sum_{i=1}^{3} e_i \otimes e_i \otimes e_i \in \mathbb{R}^{3\times 3\times 3}$$

  as $(Q, Q, Q) \cdot A \neq A$

identity tensor?

- analogue is

$$J = \sum_{i=1}^{3} \sum_{j=1}^{3} \sum_{k=1}^{3} \varepsilon_{ijk} e_i \otimes e_j \otimes e_k \in \mathbb{R}^{3\times 3\times 3}$$

where ε_{ijk} is the Levi-Civita symbol

$$\varepsilon_{ijk} = \begin{cases} +1 & \text{if } (i,j,k) = (1,2,3),(2,3,1),(3,1,2), \\ -1 & \text{if } (i,j,k) = (1,3,2),(2,1,3),(3,2,1), \\ 0 & \text{if } i = j, j = k, k = i \end{cases}$$

- $(Q, Q, Q) \cdot J = J$ for any $Q \in \mathrm{O}(3)$, unique up to scalar multiples
- J is a Cartesian 3-tensor

why important (in computations)

two simple properties:

- **group:** change-of-coordinates matrices may be multiplied/inverted:
  - if X, Y orthogonal or invertible, so is XY
  - if X orthogonal or invertible, so is X^{-1}
- **group action:** transformation rules may be composed:
  - if $a' = X^{-\mathsf{T}}a$ and $a'' = Y^{-\mathsf{T}}a'$, then $a'' = (YX)^{-\mathsf{T}}a$
  - if $A' = XAX^{-1}$ and $A'' = YA'Y^{-1}$, then $A'' = (YX)A(YX)^{-1}$

plus one more fact about the change-of-coordinate matrices (next slides)

why important (in computations)

- recall Givens rotation, Householder reflector, Gauss transform:

$$G = \begin{bmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \cdots & \cos\theta & \cdots & -\sin\theta & \cdots & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & \cdots & \sin\theta & \cdots & \cos\theta & \cdots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{bmatrix} \in \mathrm{SO}(n),$$

$$H = I - \frac{2vv^{\mathsf{T}}}{v^{\mathsf{T}}v} \in \mathrm{O}(n), \qquad M = I - ve_i^{\mathsf{T}} \in \mathrm{GL}(n)$$

- $a' = Ga$ rotation of a in (i, j)-plane by an angle θ
- $a' = Ha$ reflection of a in the hyperplane with normal $v/\|v\|$
- for judiciously chosen v, $a' = Ma \in \operatorname{span}\{e_{i+1}, \dots, e_n\}$, i.e., has $(i+1)$th through nth coordinates zero

why important (in computations)

- facts about change-of-coordinate matrices in transformation rules
  - any $X \in \mathrm{SO}(n)$ is a product of Givens rotations
  - any $X \in \mathrm{O}(n)$ is a product of Householder reflectors
  - any $X \in \mathrm{GL}(n)$ is a product of elementary matrices
  - any unit lower triangular $X \in \mathrm{GL}(n)$ is a product of Gauss transforms
- in group theoretic lingo:
  - Givens roations generate $\mathrm{SO}(n)$
  - Householder reflectors generate $\mathrm{SO}(n)$
  - elementary matrices generate $\mathrm{GL}(n)$
  - Grauss transforms generate lower unitriangular subgroup of $\mathrm{GL}(n)$

why important (in computations)

- algorithms in numerical linear algebra implicitly based on these:
  - apply a sequence of tensor transformation rules

$$A \to X_1 A \to X_2(X_1 A) \to \cdots \to B$$
$$A \to X_1^{-\mathsf{T}} A \to X_2^{-\mathsf{T}}(X_1^{-\mathsf{T}} A) \to \cdots \to B$$
$$A \to X_1 A X_1^{\mathsf{T}} \to X_2(X_1 A X_1^{\mathsf{T}}) X_2^{\mathsf{T}} \to \cdots \to B$$
$$A \to X_1 A X_1^{-1} \to X_2(X_1 A X_1^{-1}) X_2^{-1} \to \cdots \to B$$
$$A \to X_1 A Y_1^{-1} \to X_2(X_1 A Y_1^{-1}) Y_2^{-1} \to \cdots \to B$$

  - required X obtained as either $X_m X_{m-1} \dots X_1$ or its limit as $m \to \infty$
- caveat: in numerical linear algebra, we tend to view these transformation rules as giving matrix decompositions

example: full-rank least squares

- tensor transformation rules for ordinary least squares: mixed 2-tensor $A' = XAY^{-1}$ with change-of-coordinates $(X, Y) \in \mathrm{O}(m) \times \mathrm{GL}(n)$
- method of solution essentially obtains

$$X = Q \in \mathrm{O}(m), \qquad Y = R^{-1} \in \mathrm{GL}(n)$$

  by applying a sequence of tensor transformation rules
- suppose $\operatorname{rank}(A) = n$, with sequence of tensor transformation rules

$$A \to Q_1^{\mathsf{T}} A \to Q_2^{\mathsf{T}}(Q_1^{\mathsf{T}} A) \to \cdots \to Q^{\mathsf{T}} A = \begin{bmatrix} R \\ 0 \end{bmatrix}$$

  given by Householder QR algorithm, get

$$A = Q \begin{bmatrix} R \\ 0 \end{bmatrix}$$

- practically Voigt's definition: transform problem into form where solution of transformed problem is related to original solution in a definite way

example: full-rank least squares

- minimum value is invariant Cartesian 0-tensor

$$\min \|Av - b\|^2 = \min \|Q^\top (Av - b)\|^2 = \min \left\| \begin{bmatrix} R \\ 0 \end{bmatrix} v - Q^\top b \right\|^2$$

$$= \min \left\| \begin{bmatrix} R \\ 0 \end{bmatrix} v - \begin{bmatrix} c \\ d \end{bmatrix} \right\|^2 = \min \|Rv - c\|^2 + \|d\|^2 = \|d\|^2$$

where

$$Q^\top b = \begin{bmatrix} c \\ d \end{bmatrix}$$

- solution of transformed problem $Rv = c$ equals original solution, and may be obtained through back substitution, i.e., a sequence

$$c \to Y_1^{-1} c \to Y_2^{-1}(Y_1^{-1} c) \to \cdots \to R^{-1} c = v$$

where Y_i's are Gauss transforms

example: Krylov subspaces

- $A \in \mathbb{R}^{n \times n}$ with all eigenvalues distinct and nonzero, arbitrary $b \in \mathbb{R}^n$
- change-of-coordinates matrix K whose columns are

$$b, Ab, A^2 b, \ldots, A^{n-1} b$$

  is invertible, i.e., $K \in \mathrm{GL}(n)$
- transformation rule gives

$$A = K \begin{bmatrix} 0 & 0 & \cdots & 0 & -c_0 \\ 1 & 0 & \cdots & 0 & -c_1 \\ 0 & 1 & \cdots & 0 & -c_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & -c_{n-1} \end{bmatrix} K^{-1}$$

- seemingly trivial but when combined with other techniques, give powerful iterative methods for linear systems, least squares, eigenvalue problems, or evaluating various matrix functions

example: Krylov subspaces

- why not use more obvious

$$A = X \begin{bmatrix} \lambda_1 & 0 & 0 & \cdots & 0 \\ 0 & \lambda_2 & 0 & \cdots & 0 \\ 0 & 0 & \lambda_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \lambda_n \end{bmatrix} X^{-1}$$

  with change-of-coordinates matrix $X \in \mathrm{GL}(n)$ given by eigenvectors?
- much more difficult to compute than K
- one way is in fact to implicitly exploit relation between K and X:

$$\begin{bmatrix} \lambda_1 & 0 & 0 & \cdots & 0 \\ 0 & \lambda_2 & 0 & \cdots & 0 \\ 0 & 0 & \lambda_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \lambda_n \end{bmatrix} = \begin{bmatrix} 1 & \lambda_1 & \lambda_1^2 & \dots & \lambda_1^{n-1} \\ 1 & \lambda_2 & \lambda_2^2 & \dots & \lambda_2^{n-1} \\ 1 & \lambda_3 & \lambda_3^2 & \dots & \lambda_3^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \lambda_m & \lambda_m^2 & \dots & \lambda_m^{n-1} \end{bmatrix} \begin{bmatrix} 0 & 0 & \cdots & 0 & -c_0 \\ 1 & 0 & \cdots & 0 & -c_1 \\ 0 & 1 & \cdots & 0 & -c_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & -c_{n-1} \end{bmatrix} \begin{bmatrix} 1 & \lambda_1 & \lambda_1^2 & \dots & \lambda_1^{n-1} \\ 1 & \lambda_2 & \lambda_2^2 & \dots & \lambda_2^{n-1} \\ 1 & \lambda_3 & \lambda_3^2 & \dots & \lambda_3^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \lambda_m & \lambda_m^2 & \dots & \lambda_m^{n-1} \end{bmatrix}^{-1}$$

beyond linear algebra

- primal and dual forms of cone programming problem over a symmetric cone $\mathbb{K} \subseteq \mathbb{V}$ conform to transformation rules for Cartesian 0-, 1-, 2-tensors
- but change-of-coordinates matrices would have to be replaced a linear map from the *orthgonal group of the cone*:

$$\mathrm{O}(\mathbb{K}) \coloneqq \{\varphi : \mathbb{V} \to \mathbb{V} : \varphi \text{ linear, invertible, and } \varphi^* = \varphi^{-1}\}$$

- special cases include linear programming (LP), convex quadratic programming (QP), second-order cone programming (SOCP), and semidefinite programming (SDP)

beyond linear algebra

- vector space $\mathbb{V}$ may not be $\mathbb{R}^n$, e.g., $\mathbb{K} = \mathbb{S}^n_{++}$ and $\mathbb{V} = \mathbb{S}^n$ for SDP
- numerical linear algebra notations we have been using to describe definition ① awkward and unnatural
- want to work with tensors over arbitrary vector spaces
  - space of Toeplitz or Hankel or Toeplitz-plus-Hankel matrices
  - space of polynomials or differential forms or differential operators
  - space of L^2-functions on homogeneous spaces
- another impetus for coordinate-free approach in definitions ② and ③ next week

thanks

- my former postdocs Ke Ye (Chinese Academy of Sciences), Yang Qi (École Polytechnique and INRIA): taught me everything about tensors
- Keith Conrad, Shmuel Friedland, Edinah Gnang, Shenglong Hu, Risi Kondor, J. M. Landsberg, Jiawang Nie, Peter McCullagh, Emily Riehl, Thomas Schultz: learned a great deal from them too
- image credits: Alex Krizhevsky (p. 19, CIFAR-10), John Jumper (p. 19, AlphaFold 2)