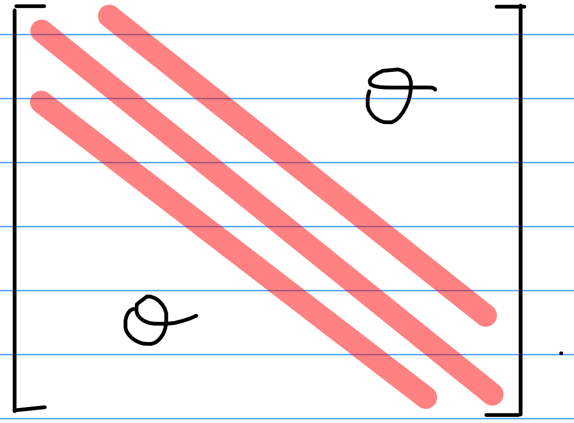
Numerical Methods for Computational Science and Engineering

Autumn Semester 2018

Prof. Rima Alaifari, SAM, ETH Zurich

2.4. Sparse Linear Systems

In different applications: LSE with sparse matrix



as system matrix

Examples: • Discretization of Poisson equation (i.e. solving PDEs)

• spline interpolation

Definition 2.5.1 (Sparse matrix). $\mathbf{A} \in \mathbb{K}^{m,n}$, $m,n \in \mathbb{N}$, is *sparse*, if

$$\underbrace{\operatorname{nnz}}_{\text{number of non-zero elements}}(\mathbf{A}) := \#\{(i,j) \in \{1,\dots,m\} \times \{1,\dots,n\} : a_{ij} \neq 0\} \underline{\ll mn}\,.$$

Ignoring sparsity: • unnecessary storage of zeros

• unnecessary computations involving zero entries

Instead: store only non-zero elements
+ "bookkeeping" i.e. keep indices of the elements

2.4.1 Sparse matrix storage formats

Goal: • required memory $\sim nnz(A)$
• cost of computing Ax $\sim nnz(A)$

How to store a sparse matrix?

Simplest idea: in "triplets"

$(0, 0, 1)$ $\qquad (A)_{1,1} = 1$

$(i, j, (A)_{i+1,j+1})$

indexing starts at 0

Then: form a vector of triplets
$\rightsquigarrow$ COO / triplet format

What if, in that list, the entries $(0,0,1)$ and $(0,0,2)$ appear?

Convention: $(A)_{11} = 1+2=3$.

Example:

$\text{triplets}(A) = ((0,2,1),(0,1,1),(1,0,1),(3,3,1),(1,0,2),(0,2,3))$

$$A = \begin{pmatrix} 0 & 1 & 4 & 0 \\ 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

in EIGEN: Triplet (individual triplet

A triplet object can be initialized as demonstrated in the following example:

Example 2.5.1:

```
unsigned int row_idx = 2;
unsigned int col_idx = 4;
double value = 2.5;
Eigen::Triplet<double> triplet(row_idx,col_idx,value);
std::cout << '(' << triplet.row() << ',' << triplet.col()
      << ',' << triplet.value() << ')' << std::endl;
```

Sparse matrices in EIGEN:

CCS/CRS format   compressed row/column format

```
#include<Eigen/Sparse>
Eigen::SparseMatrix<int, Eigen::ColMajor> Asp(rows,cols); // CCS format
Eigen::SparseMatrix<double, Eigen::RowMajor> Bsp(rows,cols); // CRS format
```

What is CRS format?

information about sparse matrix is saved in 3 arrays:

① array of nonzero entries of the matrix (length: nnz(A))
CRS ⇝ row major

② column index vector
(contains column index corresponding to each of the above entries) (length: nnz(A))

(3) "row pointer"

$row\_ptr[0] = 0$

$row\_ptr[i] = row\_ptr[i-1] + nnz((i-1)^{th} \text{ row})$

length: $m+1$ for $A \in \mathbb{K}^{m,n}$

$(row\_ptr[m+1] = nnz(A))$

Example:

$$\mathbf{A} = \begin{bmatrix} 10 & 0 & 0 & 0 & -2 & 0 \\ 3 & 9 & 0 & 0 & 0 & 3 \\ 0 & 7 & 8 & 7 & 0 & 0 \\ 3 & 0 & 8 & 7 & 5 & 0 \\ 0 & 8 & 0 & 9 & 9 & 13 \\ 0 & 4 & 0 & 0 & 2 & -1 \end{bmatrix}$$

(1) val-vector:

$[10 \quad -2 \quad \dots \quad 4 \quad 2 \quad -1]$

(2) col-ind vector:

$[0 \quad 4 \quad \dots \quad 1 \quad 4 \quad 5]$

(3) row_ptr vector:

$[0 \quad 2 \quad 5 \quad 8 \quad 12 \quad 16 \quad 19]$

Default in EIGEN: CCS

How to initialize a sparse matrix?

```
unsigned int rows, cols, nr;
.....
SparseMatrix<double, RowMajor> mat(rows, cols);
mat.reserve(RowVectorXi::Constant(rows, nr));
// do many (incremental) initializations
for ( ) {
  mat.insert(i, j) = value_ij;
  mat.coeffRef(i, j) += increment_ij;
}
mat.makeCompressed();
```

reserve ⇝ number of estimated nonzeros per row

insert ⇝ set value at (i,j) for the first time

coeffRef ⇝ update value at (i,j)

makeCompressed ⇝ to obtain CCS/CRS format (removes possible empty slots)

Note: If we have good estimate of #nnz per row (or column), then this procedure is ok

BUT: if not, we might have multiple reallocations during insertion procedure

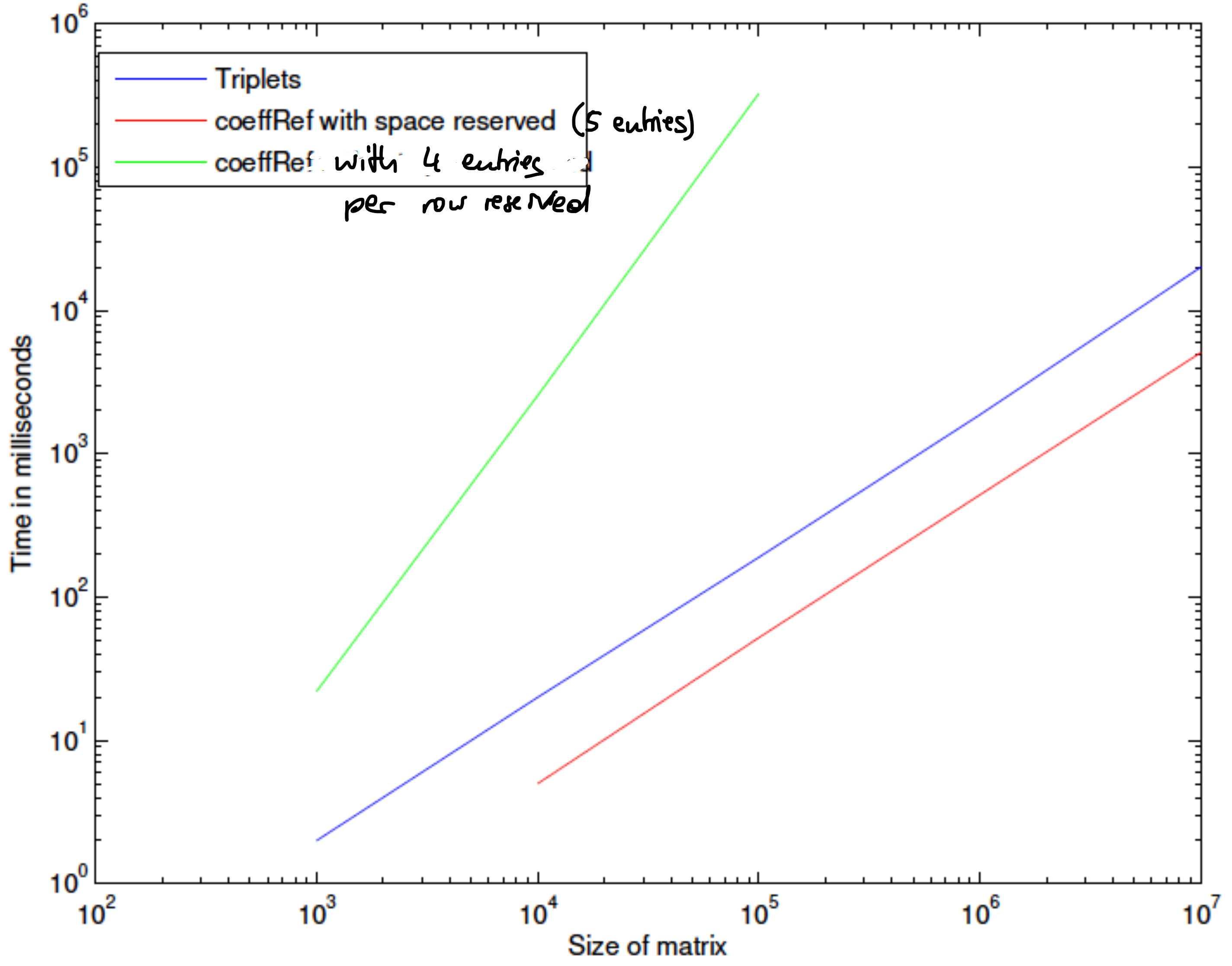
then: cost of inserting new element ~ order of current # nonzeros

Alternative:

1. Initialize with triplet format
2. Convert to CRS/CCS format

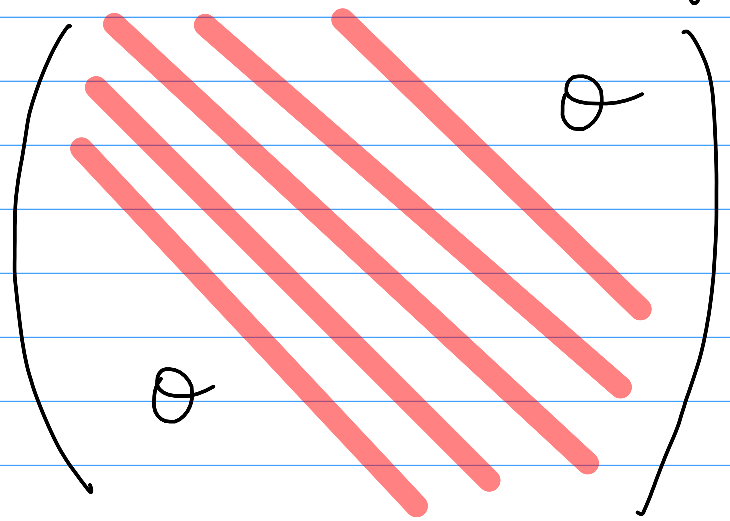
How?
- Build a vector of triplets
- then use "setFromTriplets"

```
std::vector<Eigen::Triplet<double> > triplets;
triplets.reserve(nnz);
// .. fill the std::vector triplets
Eigen::SparseMatrix<double, Eigen::RowMajor> spMat(rows,cols);
spMat.setFromTriplets(triplets.begin(),triplets.end());
```



Example : Initialization of band matrix

5 nonzero diagonals



Direct solution of sparse LSE:

built-in sparse solvers: sparse LU, sparse Cholesky factorization

take in matrices in CCS format & exploit the sparse structure

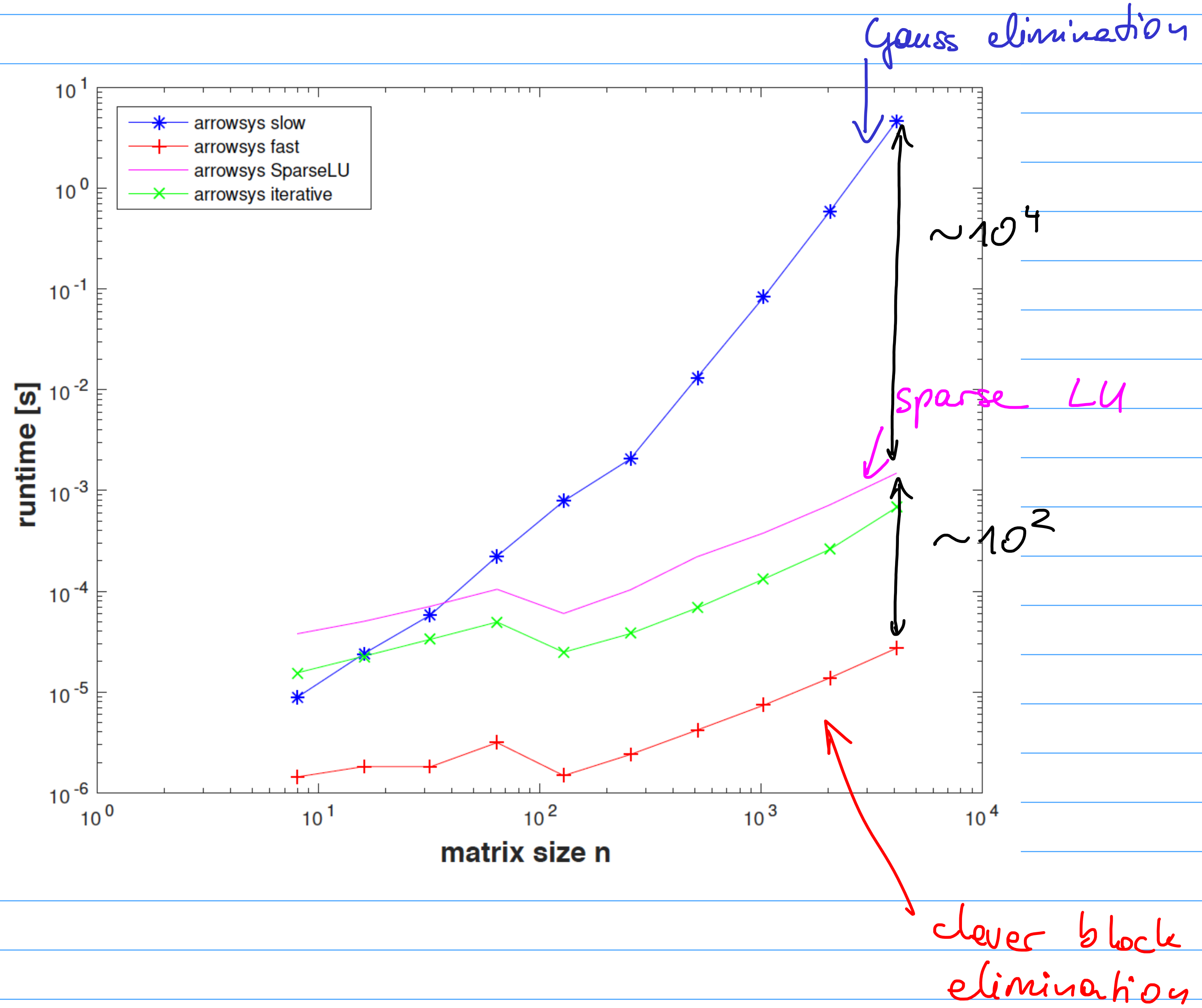
Cost of sparse solvers:

between $\mathcal{O}(nnz^{3/2})$ and $\mathcal{O}(nnz^{5/2})$

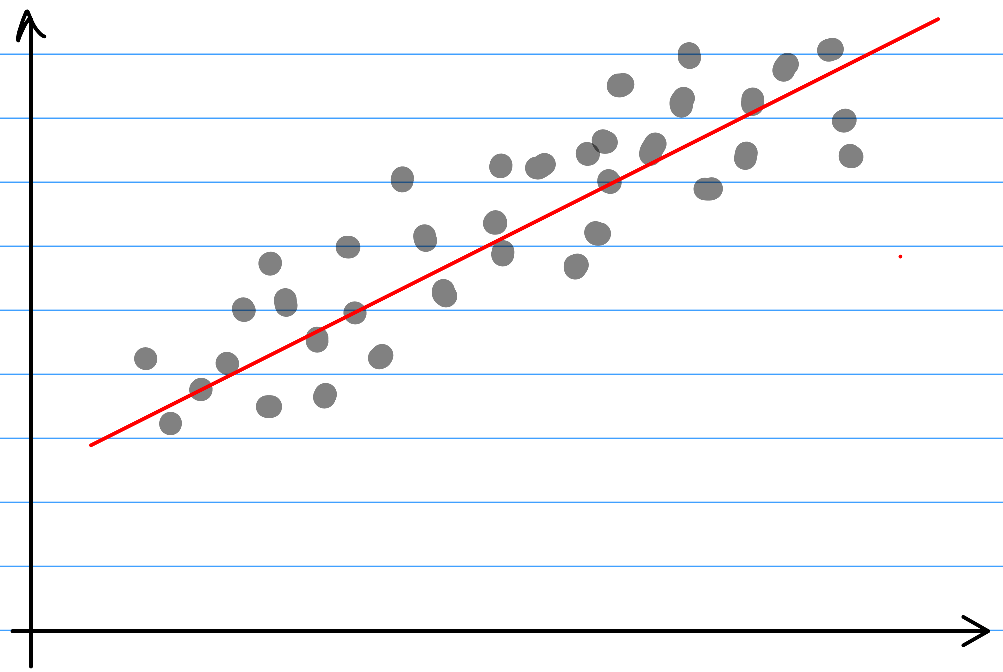
Sparse matrix solvers: very sophisticated

→ use them (don't implement yourself)

Example: LSE with arrow matrix



3. Direct methods for Linear Least Squares Problems



Linear Regression

A simple learning task

(Problem of parameter estimation)

Model: $f(x) = a_1 x_1 + \dots + a_n x_n \qquad f: \mathbb{R}^n \to \mathbb{R}$

$= \langle \vec{a}, \vec{x} \rangle$

Suppose we have series of measurements / data points

$$(x^k, y^k)_{k=1}^{m} \qquad x^k \in \mathbb{R}^n, \quad y^k \in \mathbb{R}$$

where $\quad x^k \mapsto y^k = f(x^k)$

Goal: estimate parameters $a_1, \dots, a_n$

i.e. to determine a linear model f

We can write this in matrix form:

$$\langle x^i, a \rangle = y^i$$

$$\underbrace{\begin{bmatrix} \overbrace{x_1^1 & x_2^1 & \dots & x_n^1}^{x^1} \\ x_1^2 & x_2^2 & \dots & x_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ x_1^m & x_2^m & \dots & x_n^m \end{bmatrix}}_{X} \underbrace{\begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix}}_{a} = \underbrace{\begin{bmatrix} y^1 \\ y^2 \\ \vdots \\ y^m \end{bmatrix}}_{Y}$$

In general: m much larger than n

$\uparrow$

number of data points

More generally:

$$f(x) = a_1 f_1(x) + a_2 f_2(x) + \dots + a_n f_n(x)$$

Record $(x^k, y^k)_{k=1}^{m}$ $\quad y^k = f(x^k)$

Estimate $a_1, \dots, a_n$ by solving

$$\begin{bmatrix} f_1(x^1) & f_2(x^1) & \dots & f_n(x^1) \\ f_1(x^2) & f_2(x^2) & \dots & f_n(x^2) \\ \vdots & \vdots & \ddots & \vdots \\ f_1(x^m) & f_2(x^m) & \dots & f_n(x^m) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} y^1 \\ y^2 \\ \vdots \\ y^m \end{bmatrix}$$

Example: Polynomial regression:

$$f(x) = a_1 + a_2 x + a_3 x^2 + \dots + a_n x^{n-1}$$

$$\begin{pmatrix} 1 & x^1 & (x^1)^2 & \dots & (x^1)^{n-1} \\ \vdots & \vdots & & & \\ 1 & x^m & (x^m)^2 & \dots & (x^m)^{n-1} \end{pmatrix} \begin{pmatrix} a_1 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} y^1 \\ \vdots \\ y^n \end{pmatrix}$$

Linear regression:

Overdetermined system

In general: won't have a solution

(because neither model nor measurements will be perfect)

$$(*) \quad \begin{bmatrix} \\ \\ A \\ \\ \\ \end{bmatrix} \begin{bmatrix} x \end{bmatrix} = \begin{bmatrix} \\ b \\ \\ \\ \\ \end{bmatrix}$$

$A \in \mathbb{R}^{m,n}$

$m >> n$

range $\mathcal{R}(A) = \{ y \in \mathbb{R}^m : \exists\, x \in \mathbb{R}^n \text{ s.t. } Ax = y \}$

$\dim \mathcal{R}(A) = \operatorname{rank}(A) \le n$

$\mathcal{R}(A)$ is <u>at most</u> an n-dim. subspace of $\mathbb{R}^m$

Perturbing $b \in \mathcal{R}(A)$ to b^δ:

very likely that $b^\delta \notin \mathcal{R}(A)$

$\Rightarrow$ $Ax = b^\delta$ is not solvable

Instead of solving exactly:

only for search for a good approximation s.t.

$Ax \approx b$.

More precisely: minimize norm of residual

$$\|Ax - b\|_2$$

$\rightarrow$ concept of least-squares solutions!

3.1. Least squares solutions

Definition 3.1.1 (Least squares solution). For given $\mathbf{A} \in \mathbb{K}^{m,n}$, $\mathbf{b} \in \mathbb{K}^m$ the vector $\mathbf{x} \in \mathbb{R}^n$ is a *least squares solution* of the linear system of equations $\mathbf{A}\mathbf{x} = \mathbf{b}$, if and only if

$$\mathbf{x} \in \underset{\mathbf{y}\in\mathbb{K}^n}{\operatorname{argmin}} \|\mathbf{A}\mathbf{y} - \mathbf{b}\|_2 \text{, which is equivalent to } \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2 = \inf_{\mathbf{y}\in\mathbb{K}^n} \|\mathbf{A}\mathbf{y} - \mathbf{b}\|_2 .$$

Example of parameter estimation:

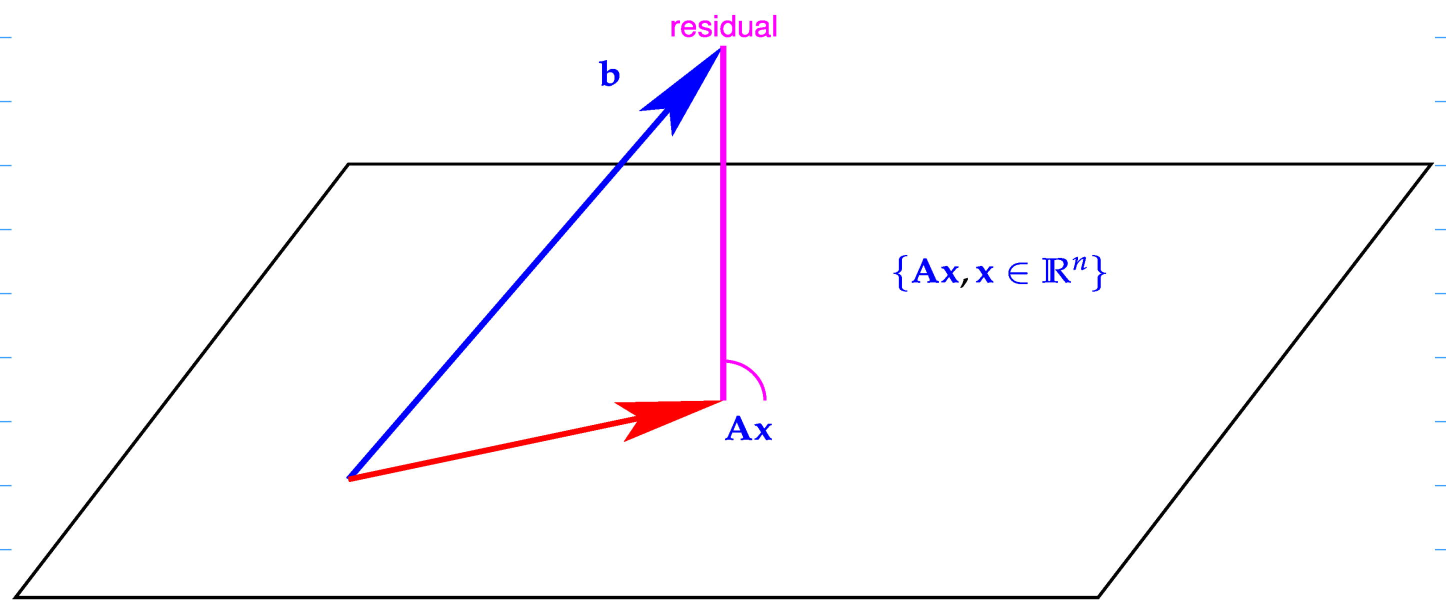
$$Xa = y$$

$$a = \underset{p \in \mathbb{R}^n}{\operatorname{argmin}} \sum_{k=1}^{m} \left| (x^k)^T \cdot p - y^k \right|^2$$

$$\operatorname{lsq}(A, b) := \{ x \in \mathbb{R}^n : x \text{ is a least-squares solution of } Ax = b \} \subset \mathbb{R}^n$$

$x \in \operatorname{lsq}(A, b)$: Ax is the closest element to b in $\mathcal{R}(A)$

i.e. projection of b on $\mathcal{R}(A)$



Theorem (Existence of least-squares solutions)

For any $A \in \mathbb{R}^{m,n}$, $b \in \mathbb{R}^m$ a least-squares solution to $Ax=b$ exists.

Recall:

Lemma: For any matrix $A \in \mathbb{K}^{m,n}$ the following holds:

$$\mathcal{N}(A) = \mathcal{R}(A^H)^\perp$$
$$\mathcal{N}(A)^\perp = \mathcal{R}(A^H)$$

$$\left[z \in Y^\perp : \langle z, y \rangle = 0 \quad \forall y \in Y \right]$$

Theorem 3.1.2 (Obtaining least squares solutions by solving normal equations) *The vector $\mathbf{x} \in \mathbb{R}^n$ is a least squares solution (see definition 3.1.1) of the linear system of equations $\mathbf{A}\mathbf{x} = \mathbf{b}$, $\mathbf{A} \in \mathbb{R}^{m,n}$, $\mathbf{b} \in \mathbb{R}^m$,* if and only if *it solves the* normal equations

$$\mathbf{A}^\top \mathbf{A}\mathbf{x} = \mathbf{A}^\top \mathbf{b} \, . \qquad (1)$$

$x \in lsq(A,b) \iff Ax$ is closest element in $\mathcal{R}(A)$ to b

$\iff Ax - b \in \mathcal{R}(A)^\perp = \mathcal{N}(A^T)$

$\iff A^T (Ax-b) = 0 \, . \qquad (1)$

$$\left[\ \mathbf{A}^\top\ \right]\left[\ \mathbf{A}\ \right]\left[\ \mathbf{x}\ \right] = \left[\ \mathbf{A}^\top\ \right]\left[\ \mathbf{b}\ \right],$$

$$\Longleftrightarrow \quad \left[\ \mathbf{A}^\top\mathbf{A}\ \right]\left[\ \mathbf{x}\ \right] = \left[\ \mathbf{A}^\top\ \right]\left[\ \mathbf{b}\ \right].$$

LSE with system matrix $\underbrace{A^T A}_{\text{symm. pos. semi-def.}} \in \mathbb{R}^{n,n}$

Uniqueness of least-squares solution??

$\hookrightarrow$ we need $\mathcal{N}(A^T A) = \{0\}$

Theorem: For any matrix $A \in \mathbb{R}^{m,n}$, $m \geq n$:

$$\mathcal{N}(A^T A) = \mathcal{N}(A)$$
$$\mathcal{R}(A^T A) = \mathcal{R}(A^T).$$

For uniqueness of least-sq. solution, we need

$\mathcal{N}(A) = \{0\} \iff \operatorname{rank}(A) = n$

Corollary 3.1.1 (Uniqueness of least squares solutions). *If $m \geq n$ and* $\cancel{\mathrm{Ker}}\,\mathcal{N}(\mathbf{A}) = \{\mathbf{0}\}$*, then the linear system of equations $\mathbf{A}\mathbf{x} = \mathbf{b}$, $\mathbf{A} \in \mathbb{R}^{m,n}$, $\mathbf{b} \in \mathbb{R}^m$, has a unique least squares solution (see definition 3.1.1)*

$$\mathbf{x} = (\mathbf{A}^\top\mathbf{A})^{-1}\mathbf{A}^\top\mathbf{b}\,,$$

that can be obtained by solving the normal equations (3.6).

3.1.1 Generalized solutions & Moore-Penrose Pseudoinverse

How to overcome possible non-uniqueness:

$\rightarrow$ pick least-sq. solutions with minimal norm.

Definition 3.1.2 (Generalized solution of a linear system of equations). The *generalized solution* $\mathbf{x}^\dagger \in \mathbb{R}^n$ of a linear system of equations $\mathbf{A}\mathbf{x} = \mathbf{b}$, $\mathbf{A} \in \mathbb{R}^{m,n}$, $\mathbf{b} \in \mathbb{R}^m$, is defined as

$$\mathbf{x}^\dagger := \operatorname{argmin}\{\|\mathbf{x}\|_2 : \mathbf{x} \in \operatorname{lsq}(\mathbf{A}, \mathbf{b})\}\,. \tag{1}$$

Theorem: For $A \in \mathbb{R}^{m,n}$, $b \in \mathbb{R}^m$ the generalized solution to $Ax = b$ is unique.

Why? Suppose $x^+ = x_1 + x_2$ $\quad x_1 \in \mathcal{N}(A)^\perp$

$x_2 \in \mathcal{N}(A)$

$\mathbb{R}^n = \mathcal{N}(A) \oplus \mathcal{N}(A)^\perp$

$$\|x^+\|_2^2 = \|x_1\|_2^2 + \|x_2\|_2^2$$

$$A^T(Ax^+ - b) = 0$$

$$A^T(Ax_1 + \underbrace{Ax_2}_{=0} - b) = 0$$

$$A^T(Ax_1 - b) = 0$$

$$\uparrow$$

$$x_1 \in \operatorname{lsq}(A, b)$$

$$\|x_1\|_2 \le \|x^+\|_2 = \min \|Ax - b\|_2$$

$$\Rightarrow \underline{x^+ \in \mathcal{N}(A)^\perp}$$

Uniqueness: ① $x_1^+, x_2^+ \in \mathcal{N}(A)^\perp \Rightarrow \underline{x_1^+ - x_2^+ \in \mathcal{N}(A)^\perp}$

② $A^T A(x_1^+ - x_2^+) = 0 \Rightarrow x_1^+ - x_2^+ \in \mathcal{N}(A^T A) = \underline{\mathcal{N}(A)}$

$$\underline{\underline{x_1^+ = x_2^+}}.$$

Formula for generalized $x^+ \in \mathcal{N}(A)^\perp$.

Idea: Find a basis $\{v_1, \dots, v_k\} \subset \mathbb{R}^n$ of $\mathcal{N}(A)^\perp$

$k = \dim \mathcal{N}(A)^\perp$

and write x^+ in this basis:

$$x^+ = \underbrace{y_1}_{\text{coefficients}} v_1 + \dots + \underbrace{y_k}_{} v_k$$

$$V = \begin{bmatrix} | & & | \\ v_1 & \dots & v_k \\ | & & | \end{bmatrix}$$

One can always find a vector y s.t.

$$Vy = x^+$$

$$A^T A x^+ = A^T b$$

$$V^T A^T A V y = V^T A^T b$$

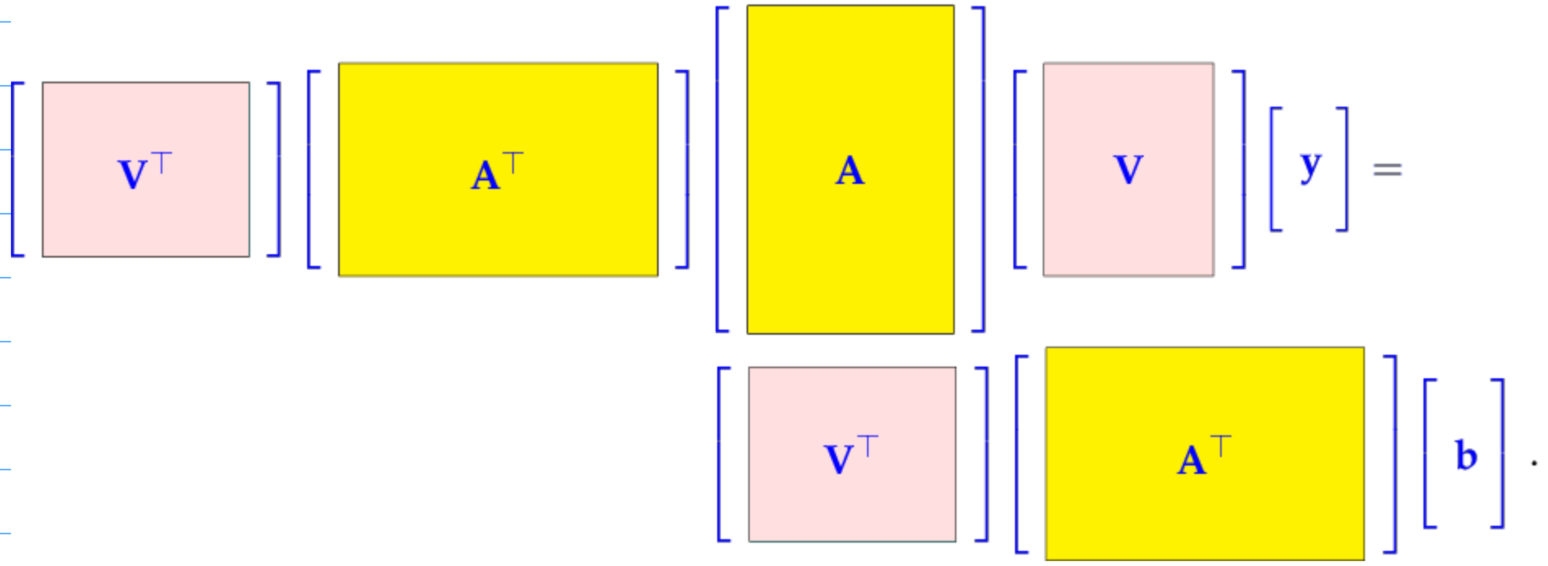
(reduced normal equations)

It means: We search for a solution to the normal equation in $\mathcal{N}(A)^\perp$.

(because $\mathcal{N}(A) = \mathcal{N}(A^T A)$ is the source of nonuniqueness).

$$\mathbf{V}^\top\mathbf{A}^\top\mathbf{A}\mathbf{V}\,\mathbf{y} = \mathbf{V}^\top\mathbf{A}^\top\mathbf{b} \tag{3.1.36}$$

$$\Updownarrow$$



By construction: $\mathcal{N}(AV) = \{0\}$

$\Rightarrow \mathcal{N}(V^\top A^\top A V) = \{0\}$

$\Rightarrow$ unique solvability of $V^\top A^\top A V y = V^\top A^\top b$.

Get x^+ by computing $x^+ = Vy$.

Theorem 3.1.5 (Formula for generalized solution)

Given $\mathbf{A} \in \mathbb{R}^{m,n}$, $\mathbf{b} \in \mathbb{R}^m$, *the generalized solution* $\mathbf{x}^\dagger$ *of the linear system of equations* $\mathbf{A}\mathbf{x} = \mathbf{b}$ *is given by*

$$\mathbf{x}^\dagger = \mathbf{V}(\mathbf{V}^\top\mathbf{A}^\top\mathbf{A}\mathbf{V})^{-1}(\mathbf{V}^\top\mathbf{A}^\top\mathbf{b}) ,$$

where $\mathbf{V}$ *is* any *matrix whose columns form a basis of* $\mathrm{Kern}(\mathbf{A})^\perp$.

$V(V^\top A^\top A V)^{-1} V^\top A^\top$ is called the Moore-Penrose Pseudoinverse A^+ of A.

A^+ does not depend on the choice of V.

3.2. Normal Equation Methods

Suppose we have $A \in \mathbb{R}^{m,n}$, $m > n$, with full rank ($\operatorname{rank}(A) = n$).

❶ Compute regular matrix $\mathbf{C} := \mathbf{A}^\top \mathbf{A} \in \mathbb{R}^{n,n}$

❷ Compute right hand side vector $\mathbf{c} := \mathbf{A}^\top \mathbf{b}$

❸ Solve symmetric positive definite linear system of equations: $\mathbf{C}\mathbf{x} = \mathbf{c}$

$$\left.\begin{array}{ll} \text{step ❶:} & \text{cost } \mathcal{O}(mn^2) \\ \text{step ❷:} & \text{cost } \mathcal{O}(nm) \\ \text{step ❸:} & \text{cost } \mathcal{O}(n^3) \end{array}\right\} \Longrightarrow \quad \text{cost } \mathcal{O}(n^2 m + n^3) \quad \text{for } m, n \to \infty\,.$$

C is pos. def. $\quad x^T C x = x^T A^T A x = \|Ax\|^2 > 0$ for all $x \neq 0$.

Note on stability:

$$\operatorname{cond}(A^T A) = \operatorname{cond}(A)^2$$

condition number squares!

Example: $\quad A = \begin{bmatrix} 1 & 1 \\ \delta & 0 \\ 0 & \delta \end{bmatrix} \qquad \delta = \frac{\sqrt{EPS}}{2}$

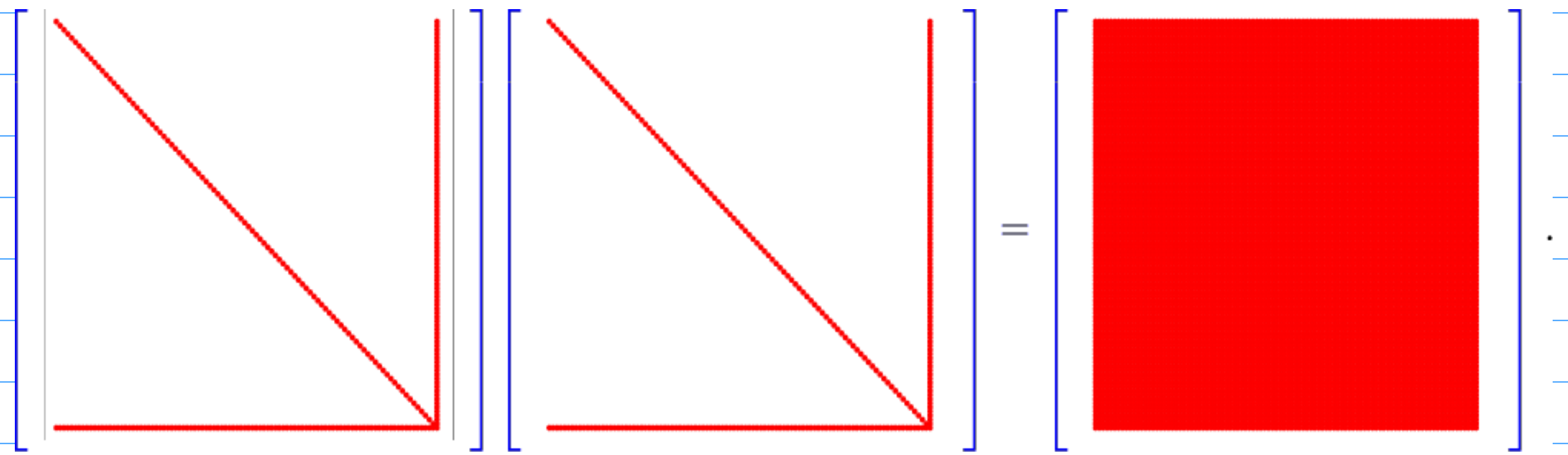
$$A^T A = \begin{bmatrix} 1+\delta^2 & 1 \\ 1 & 1+\delta^2 \end{bmatrix} \qquad 1+\delta^2 = 1 + \frac{EPS}{4} \underset{\uparrow}{=} 1$$

in machine numbers

As an element of $\mathbb{M}^{2,2}$ $A^T A$ is not regular.

Further note:

A sparse $\not\Rightarrow$ $A^T A$ sparse



arrow matrix: A sparse, but $A^T A$ not sparse

$\left.\begin{array}{l} \text{① Squaring cond. number} \\ \text{② Loss of sparsity} \end{array}\right\}$ challenges

Easy fix for maintaining sparsity:

Rewrite normal equations as:

(I) $\quad Ax - b = r$

(II) $\quad A^H r = 0$

$$A^H A x = A^H b \quad \Leftrightarrow \quad B \begin{bmatrix} r \\ x \end{bmatrix} := \begin{bmatrix} -I & A \\ A^H & 0 \end{bmatrix} \begin{bmatrix} r \\ x \end{bmatrix} = \begin{bmatrix} b \\ 0 \end{bmatrix}$$

if A is sparse $\Rightarrow$ B is sparse

BUT: conditioning has not improved

More generally: $r := \alpha^{-1}(Ax-b)$

for some choice of parameter α

$$\mathbf{A}^H\mathbf{A}\mathbf{x} = \mathbf{A}^H\mathbf{b} \quad \Leftrightarrow \quad \mathbf{B}_\alpha \begin{bmatrix} \mathbf{r} \\ \mathbf{x} \end{bmatrix} := \begin{bmatrix} -\alpha\mathbf{I} & \mathbf{A} \\ \mathbf{A}^H & 0 \end{bmatrix} \begin{bmatrix} \mathbf{r} \\ \mathbf{x} \end{bmatrix} = \begin{bmatrix} \mathbf{b} \\ 0 \end{bmatrix}.$$

hope: $\operatorname{cond}(B_\alpha) \approx \operatorname{cond}(A)$

3.3. Orthogonal Transformation Methods

Consider least-squares problem $Ax=b$ $A \in \mathbb{R}^{m,n}$

$m > n$

$\operatorname{rank}(A) = n$

Idea: Instead of solving $Ax=b$

find $\tilde{A}, \tilde{b}$

with $\operatorname{lsq}(A,b) = \operatorname{lsq}(\tilde{A}, \tilde{b})$ s.t.

$\tilde{A}x = \tilde{b}$ is easier to solve

(triangular system)

Consider unitary (or orthogonal) matrices:

unitary: $Q \in \mathbb{K}^{n,n}$: $Q^{-1} = Q^H$

orthogonal: $Q \in \mathbb{R}^{n,n}$: $Q^{-1} = Q^T$

i.e. $Q^H Q = I = Q Q^H$

Q is unitary / orth. if & only if

$\|Qy\|_2 = \|y\|_2 \quad \forall\, y \in \mathbb{R}^n.$

i.e. Q preserves the norm.

Now: For least-squares solution

$$\operatorname{lsq}(A, b) = \operatorname{lsq}(Q^T A, Q^T b)$$

Why:

$$x \in \operatorname{lsq}(Q^T A, Q^T b) \Leftrightarrow$$

$$A^T \underbrace{Q Q^T}_{I} A x = A^T \underbrace{Q Q^T}_{I} b$$

$$\Leftrightarrow \quad A^T A x = A^T b \quad \Leftrightarrow \quad x \in \operatorname{lsq}(A, b).$$

We conclude:

Problems $Ax = b$ and

$$Q^T A x = Q^T b$$

are equivalent in the least-squares sense.

Goal: Shape of $Q^T A$ is upper triangular.

$\rightarrow$ QR factorization

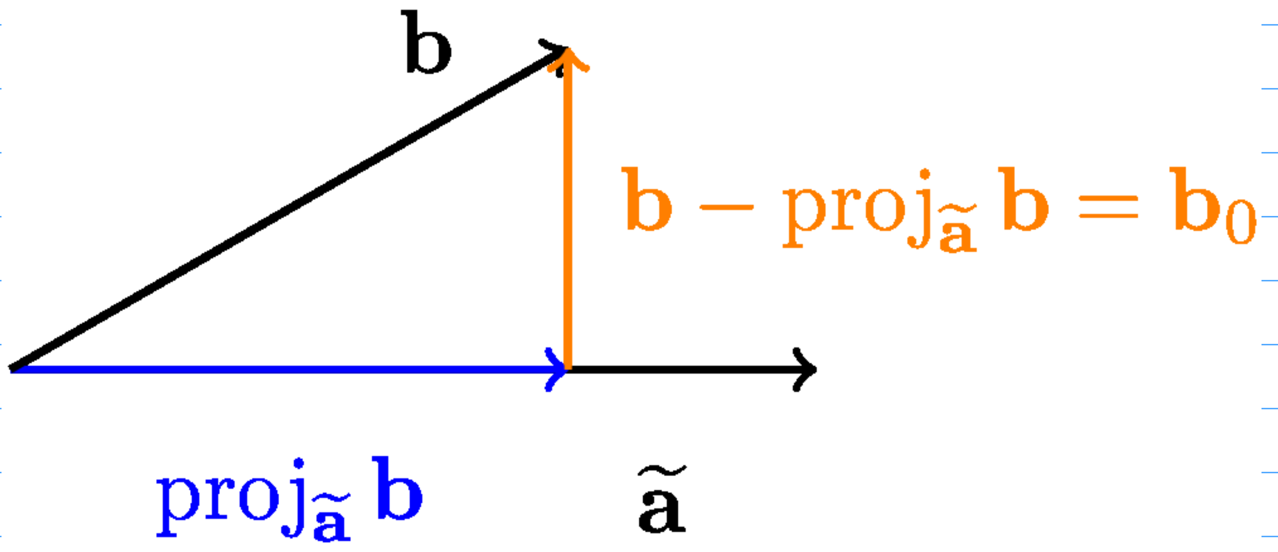
$$QR = A \quad \Leftrightarrow \quad R = Q^T A$$

$Rx = Q^T b$ is easy (back substitution).

3.3.1 QR Decomposition

First approach: Gram–Schmidt orthogonalization

Orthogonalization: suppose we have 2 linearly independent vectors $a, b \in \mathbb{R}^m$



$$\tilde{a} := \frac{a}{\|a\|_2}$$

$\mathrm{proj}_{\tilde{a}}\, b$ formulate as least-sq. problem

find τ that minimizes $\|\tau \cdot \tilde{a} - b\|_2$

$$\Leftrightarrow \quad \underbrace{\tilde{a}^T \tilde{a}}_{=1} \tau - \tilde{a}^T b = 0$$

$$\tau = \langle \tilde{a}, b \rangle$$

$$b_0 = b - \mathrm{proj}_{\tilde{a}}\, b = b - \langle a, b \rangle \frac{a}{\|a\|_2^2}$$

$$\tilde{b} := \frac{b_0}{\|b_0\|_2}$$

$\mathbf{q}^1 := \frac{\mathbf{a}^1}{\|\mathbf{a}^1\|_2}$ *% 1st output vector*
for $j = 2, \ldots, k$ **do**
  { *% Orthogonal projection*
  $\mathbf{q}^j := \mathbf{a}^j$
  for $\ell = 1, 2, \ldots, j-1$ **do**
    { $\mathbf{q}^j \leftarrow \mathbf{q}^j - \langle \mathbf{a}^j, \mathbf{q}^\ell \rangle\, \mathbf{q}^\ell$ }
  if ($\mathbf{q}^j = \mathbf{0}$) **then** *STOP*
  else { $\mathbf{q}^j \leftarrow \frac{\mathbf{q}^j}{\|\mathbf{q}^j\|_2}$ }
}

Theorem 3.3.2 (Span property of G.S. vectors)
If $\{\mathbf{a}^1, \ldots, \mathbf{a}^k\}$ is linearly independent, then the Gram-Schmidt algorithm computes orthonormal vectors $\mathbf{q}^1, \ldots, \mathbf{q}^k$ satisfying

$$\mathrm{Span}\{\mathbf{q}^1, \ldots, \mathbf{q}^\ell\} = \mathrm{Span}\{\mathbf{a}^1, \ldots, \mathbf{a}^\ell\},$$

for all $\ell \in \{1, \ldots, k\}$.

Example: $a_1 = \begin{pmatrix}1\\1\end{pmatrix}$ $\quad a_2 = \begin{pmatrix}1+\varepsilon\\1\end{pmatrix}$ $\quad \varepsilon \ll 1$

$$\mathrm{span}\{a_1, a_2\} = \mathrm{span}\left\{\begin{pmatrix}1\\0\end{pmatrix}, \begin{pmatrix}0\\1\end{pmatrix}\right\}$$

Gram-Schmidt gives:

$$q_1 = \frac{1}{\sqrt{2}}\begin{pmatrix}1\\1\end{pmatrix} \qquad \mathrm{proj}_{q_1} q_2 = \frac{2+\varepsilon}{2}\begin{pmatrix}1\\1\end{pmatrix}$$

$$\tilde{q}_2 = \begin{pmatrix}1+\varepsilon\\1\end{pmatrix} - \frac{2+\varepsilon}{2}\begin{pmatrix}1\\1\end{pmatrix} = \frac{\varepsilon}{2}\begin{pmatrix}1\\-1\end{pmatrix}$$

$$\|\tilde{q}_2\|_2 = \frac{\varepsilon}{\sqrt{2}} \qquad q_2 = \frac{\tilde{q}_2}{\|\tilde{q}_2\|_2}$$

division by $\varepsilon \ll 1$.

Can we find numerically stable QR decomposition?

Computation of QR decomposition

$$Q_n \cdot \ldots \cdot Q_2 \cdot Q_1 A = R$$

orthogonal

upper triangular